

A large, stylized letter 'A' in blue and black, positioned on the left side of the cover, partially overlapping the title area.

April 1990

Vol. 3

Nº 7

Price £2.00

Archive

The Subscription Magazine for Archimedes Users

A large, stylized letter 'E' in blue, positioned on the left side of the cover, partially overlapping the article titles.

Using the PC Emulator – Part 1

Choosing a Laser Printer


Writing RISC-OS Applications – 3

Long Integer Arithmetic / Calculating Primes

Teletext / Z88 File Transfer / Intro to C – Part 6

Reviews: Impression, U.I.M., COPS

Plus regular features – SCSI, Hardware, PipeLine,
DTP, FWP, etc, etc.

A solid blue horizontal bar at the bottom of the page.

Ram prices rising?

After my statements last month about a potential rise in the price of ram, there has been no sign of it yet. Indeed, I have just spoken to one hardware manufacturer who suggests that there may not be a price rise after all! So, before anyone accuses me of trying to scare them into buying memory upgrades, I should tell you that I believed it and, to try to hold the prices down for you, bought well over 100 Mbyte of A410 ram! If the price drops, I'm in trouble! Such is life...

Brighten up the office

The Archive office is looking a bit dull at the moment – I have just taken down the wonderful array of Christmas cards you sent us! I left them up because it brightens the place up a bit. When I realised that there were only 234 shopping days to next Christmas, I thought they had better come down.

How about brightening up the office for us? It would be great if some of you could send us a postcard of where you live – we could put up a world map and a big UK map to put your cards on. If we get too many, I'm sure one of the charities would be able to benefit from them.

Charity projects

Talking of charities, I asked if anyone objected to us sending money to the ITV Telethon. Only one objected and two or three said they were quite happy for me to choose which charities I gave the money to. I took that to mean that the silent majority thought it was OK so I have sponsored a couple of Telethon's fund raising ideas: we are going to pay about £500 for the materials for 500 first day covers to be signed by Jimmy Saville(!) and sold for £5 each. If you want one, write to Gail Halley, Telethon, Anglia House, Norwich, NR1 3JG. Secondly, we're paying for the petrol (about another £500) for four local prison officers to drive to every prison in England and Wales as a competition to guess the total distance travelled. They hope to collect between £20,000 and £50,000. That makes a total of about £1,000 out of the £13,000+ that we have raised so far.

Please keep the charity donations and charity software coming!

Many thanks,



Government Health Warning: Reading this may seriously affect your spiritual health.

OK, so why do Christians wear a gallows as a symbol of love and peace and why do we place such importance on the death of Jesus? How can I explain it in a few short sentences without trivialising it? Dunno, but here goes...

(1) God made the world – absolutely perfect it was! (2) He gave man freedom of choice – how else could we respond to His love? You can't get a robot to love you! (3) We decided we could cope without God. "We don't need you, God!"... just look around and see what a mess we made as a result! (4) (and this is the tough bit...) God said, "If you reject me, even though I made you, I will reject you." Sorry, but that's what the bible says, not my idea. Read 2 Thessalonians chap. 1, vv 8 & 9 if you can take it neat. Thankfully, that is not the end of it... (5) God sent His Son Jesus to earth – He was perfect and yet He was killed – worse than that – on the cross He was separated from His Father. "My God, my God why have you forsaken me?" This is the crunch – we deserved to be rejected, not Him, but He died and was rejected **on our behalf**. (6) The resurrection proves that His sacrifice was accepted by God and we can be 'born again' to a new life – free and forgiven! Now THAT, if it is true, is GOOD NEWS! !

Sorry that was a bit concise. If you are interested, I have a few copies of a book that will help to explain it better. Drop me a line if you would like to enquire further. Happy Easter!!

Archive

Volume 3 • Nº 7 • April 1990

Contents

Products Available	2	PipeLine	35
Hints & Tips	5	Choosing a Laser – HP Option ...	38
Comment Column	13	Calculating Prime Numbers	41
Help Needed & Offered	17	Languages Forum	44
Small Ad's	17	Arithmetic with Long Integers....	46
Hardware Column	19	COPS – Adventure Game	48
SCSI Column	20	Writing RISC-OS Applic'ns – 3	50
Ultra Intelligent Machine?	21	Teletext on the Archimedes.....	53
MS-DOS Column	23	Z88 File Transfer	56
Competition Corner	25	Using the PC Emulator – 1	59
DTP Column	26	Contact Box	60
First Word Plus Column.....	28	Fact-File	61
Introduction to C – Part 6	32		

Products Available

• **1 Gigabyte Tapestreamer!** – Oak Computers are now selling a 1Gbyte (1000 Mbyte) Tape-streamer for just £3999 + VAT (£4030 inclusive from Archive). For more details, see the SCSI Column on page 20.

• **A480's are here!** – By the time you read this it should be possible to buy from Atomwide (£850 + VAT) or through Archive (approx £890 inclusive) a 4 to 8 Mbyte upgrade for A440's and A440/1 (or upgraded 410's and 420's). The reason that 300 series computers are excluded is that it uses the co-processor slot on the backplane to get at all the necessary address lines. One board therefore plugs into the backplane and the other connects to the existing MEMC socket. The board should be low consumption because it uses eight 4Mbit chips to provide the extra 4 Mbytes instead of the thirty-two 1Mbit chips making up the existing 4 Mbytes. The boards are "right way up", unlike some other memory up-grades, as this aids air flow and reduces the likelihood of over-heating.

• **Better Multi-sync at lower price** – Eizo used to supply a low-radiation, low static multi-sync monitor (the 9060SZ) at about £40 more than the 9060S. They have now discontinued the 9060S and brought the price of the SZ down to the same price. Also, because of larger quantities that we are now buying, we have been able to increase the discount. The 9060SZ is now available at £530 (or £510 if you buy it at the same time as an A410 or A3000).

• **Careware N°6** contains Application maker, Scientific calculator, StickyBD, Sparkplug, Desktop utilities, Wordpro <-> FWPlus converter, RFSmod for CC ROM podule, 17 Maestro files (ancient & modern), Hangman (very graphic!), Connect 5 (one-up-manship?) & Polymos (based on ancient chinese game).

• **Shareware N°7 updated** – Thanks to Eric Ayers, Shareware 7 is now updated to volume 3, issue 6. Also, the data is now compatible with Arcscan II (£17 through Archive). We will put the data on this month's program disc and on the Archive bulletin board. We cannot update old Shareware 7 discs free of charge because of the huge amount of admin involved if several hundred readers sent their discs

in for update. Either send us a donation to charity when you send in your old S/W7 for updating, or just order a new S/W7 at £3 – hardly an excessive charge if you add up the cost of the disc plus postage and packing plus an admin charge!

• **Shareware N°18** – (which I forgot to mention last month) contains various !Draw files containing 3 drawn fonts and a number of coloured pictures, several demo programs, space battles, rocks (asteroids), Maestro files (classical & modern) and Imperial College's HOPE language interpreter.

• **Shareware N°23** – contains SoundTracker with tunes, Address book, File loading utility, Text printer, SetType, Sparkplug, Using View on RISC-OS, Pelmanism, Connect 4, 3 demos: Baby blues, Scrolling text & Bouncing Bars.

• **Cheaper ST506 drives for A410** – We are now stocking Oak Computers ST506 drives for 400/1 series computers. They are slightly slower access speeds (28 ms) than the Computerware ones (22 or 24 ms) but also slightly cheaper: £210, £330 and £410 for 20M, 40M and 47M respectively as compared to £230, £420 and £510.

• **Colour Digitiser from Wild Vision** – a new colour digitiser including "FastGrab" software is now available from Wild Vision at an introductory price of £339 + VAT (or £360 inclusive through Archive). We have had a look at it and it seems very impressive especially at the introductory price. This price is to be reviewed at the end of April. More details are available in the comment column on page 13.

• **More DTP Images** – G.A.Herdman Educational Software are increasing their range of discs of images for use in DTP to five at £8.99 each or various discount prices for different numbers of discs.

• **GINO is here!** – Fortran users will be pleased to hear that Intelligent Interfaces have produced GINO (£750 + VAT) in the form of three modules: GINO-F 3D which is a library of over 200 user-callable Fortran subroutines including basic line and arc drawing, characters and symbols, colour, line styles, full 3D transformations, windowing, projections and viewing; GINOGRAPH which provides advanced facilities for producing graphs,

histograms and pie charts by simple automatic call instructions or more complex and flexible routines; and GINOSURF which displays 3D data as contour maps or surface views with hidden line removal.

• **Impression with no dongle?** – In response to popular demand, Computer Concepts have decided that it will release an educational network version of Impression that does not require a dongle. This version will be licenced to the particular school and will only run on their networked machines. (This is separate from, but complementary to, the existing offer of conventional 'dongled' Impressions where schools buying 10 or more copies pay only £60 + VAT per copy.)

• **Last days of Doom** – £19.95 (or £19 through Archive) from Topologika – over 150k of text and 218 'rooms'. This is the third part of Peter Killworth's Doom adventure trilogy.

• **Maps of the World** – £29.95 (inc VAT & carriage) (£28 through Archive) from Micro Studio is a graphics library pack for the Archimedes featuring over 150 maps of various countries of the world.

• **Personal Accounts** – a new program from Apricote Studios at just £14.95 is aimed at individual or club accounts. It features multiple bank accounts, direct debits and a range of editing and searching facilities. The number of entries is limited only by memory capacity e.g. 1 Mbyte A3000 would give you over 20,000 entries. (If it is as good as their Account Book program or their Invoice Program then I suspect they have under-priced it at £14.95. I hope to have a review copy soon.)

• **Prices down again!** – In line with our general policy of keeping prices down as much as possible, Norwich Computer Services have reduced the prices of binders, for example, by a whole 50p!!! Seriously though, what we have done is to try to make things simpler for ourselves (and for you) by making all our prices multiples of £1. This means that most of the items on the price list that were £n.50 are now £n.

• **Rotor – Getting in a Spin** – a new game from a new supplier, Active Sales, is available for £24.99 (or £23 through Archive). Active are not new to the games scene having started on Amiga and ST but are now moving into the Archimedes area. Their first offering is a very intricate looking space maze

game, a little like Caverns from Minerva Software.

• **VIDC enhancer** from Atomwide (£30 +VAT or £35 through Archive) is a hardware modification which increases the speed of the clock on the VIDC so that you can increase the amount of information displayed on the screen. Basically it will give you a range of extra modes in 2, 4 or 8 colours, up to the same resolution as the high res mono output on the A400 series machines.

• **!Works #1** – Jim Markland has produced a programmers's toolbox to enable you to output DXF files compatible with !Draw. These files can be read by humans(!), merged, edited etc and ported across to other computers. £10 from Jim Markland.

Review Software Received...

Apart from reviews already written, we have received review copies of the following software: 'Micro Trader' from Meadow Computers, The Account Book + The Invoice Program from Apricote Studios, Chaos (needs Logotron Logo), Armatron game, Holed Out Designer.

FUTURE Software

MINDWARP: *This game takes space invaders into the 21st Century. Over 40 levels of superbly animated attack waves. Too many original game-play features to list. (£14.95)*

COBRA: *The old snake game classic. (See review in past issue of Archive)(£5.95)*

NB: Both games require at least 1Mb of memory and RISCOS.

**Please make all cheques/P.O.s payable to
R.Millican.**

***Future Software, 10 Stokesay,
Bidston, Birkenhead,
Merseyside, L43 7PU.***

16 bit SCSI



Oak's high speed 16 bit SCSI module offers a new level of performance for the entire Archimedes range (including the A3000) with data transfer rates of up to 1.4Mb per second, a considerable increase over both ST306 and 8 bit SCSI controllers. Up to 2 Gigabytes of winchester storage may be fitted per machine, and seven SCSI devices including 4 winchesters may be attached to the card.

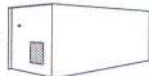
The card provides a new filing system 'SCSIFS', and an icon and filer for the Risc OS desktop. It can work in tandem with ADFS winchesters, and is compatible with the PC emulator. Low level (SWI) support is provided for other SCSI devices (e.g. tape streamers, CD ROMS etc.)



Internal winchester kits are supplied as 'plug in and go' units, with all necessary cabling and mounting hardware, and external drives are supplied in sturdy metal cases, colour matched to the Archimedes, and have their own power supply and fan. External drives are also suitable for the A3000.

A comprehensive manual with easy to follow fitting instructions is provided with each system, along with a versatile formatting and utility program. Oak SCSI drives may be write protected for security.

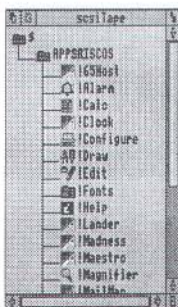
All Oak SCSI winchester drives are subjected to rigorous quality control procedures, and each drive comes with its own test certificate.



Internal Drives (inc. SCSI card)		External Drives (inc. SCSI card)	
20Mb (HDINARC20SC)	£375.00	20Mb (HDEXARC20SC)	£535.00
45Mb (HDINARC45SC)	£495.00	45Mb (HDEXARC45SC)	£655.00
80Mb (HDINARC80SC)	£795.00	80Mb (HDEXARC80SC)	£955.00
100Mb (HDINARC100SC)	£1114.00	100Mb (HDEXARC100SC)	£1274.00
200Mb (HDINARC200SC)	£1458.00	200Mb (HDEXARC200SC)	£1618.00
SCSI Card (SCSIARC)	£199.00	100Mb (HDEXARC300SC)	£2845.00
SCSI Card (SCSI3000)	£199.00	P&P external drives	£15.00
P&P internal drives/card	£10.00		

16 bit SCSI Controller Card and High Speed Winchester Drives

Tape Backup



Oak SCSI tape streamers, available in 60Mb and 150Mb capacities are the ideal means of backing up large amounts of data. Using 'DC600' type data cartridges and high quality tape drive units, Oak tape streamers provide a reliable insurance against data loss. Multi-tasking, window software allows information to be backed up from any Archimedes filing system. Restoring data from tape is simple. A tree viewer of the directory structure stored on the tape, or a 'filer' type

display can be shown in a window, and then the files to be restored can be simply 'dragged' with the mouse to the destination filing system, or even into an application!

Unattended backups may be triggered on a regular basis using the inbuilt automatic backup facility. Prompts to prevent accidental overwriting of existing files may be given during a restore as required.

Note: An Oak SCSI controller card is required. Drives are supplied with 1 free tape cartridge.



Tape Streamers	
60Mb without SCSI card (TS60SCA)	£999.95
60 Mb with SCSI card (TS60SC)	£1099.95
150Mb without SCSI card (TS150SCA)	£1254.00
150Mb with SCSI card (TS150SC)	£1354.00
P&P on Tape Streamers	£15.00
60Mb Tape Cartridge	£24.95
150Mb Tape Cartridge	£27.95
P&P on Tape Cartridges	£0.75

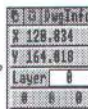
SCSI Tape Streamer

2D Draughting



Worra CAD is a new, fully-featured co-ordinate based draughting package written by the team which produced Oak PDT. Worra CAD makes full use of the multi-tasking Risc OS environment, and in doing so provides an exceptionally easy to learn and use, yet tremendously powerful and productive user interface. 'Hot Keys' further enhance the program's productivity for the experienced user. The program can handle multiple drawings, and can display multiple views of each drawing.

Worra CAD uses extended precision floating point arithmetic, accurate to 18 significant figures, to ensure that all geometry is as accurate as possible, and allows Worra CAD to be used where integer packages simply would not provide the precision required.



Standard features include 16 layers, hatching, user-definable grid, 16 million colours, automatic associative dimensioning and a wide range of geometric construction and object snap facilities. Drawings can be exported to other packages (e.g. 'Draw and then on to DTP etc.) via DXF files, or transferred as sprites to art packages. Hard copy can be produced from either the HPGL plotter output or via Risc OS printer drivers.

Worra CAD costs only **£75.00** (+£5.00 p&p)

WorraCAD



Oak Computers
Cross Park House
Low Green
Rawdon
Leeds
LS19 6HA
Tel: 0532 502615
Fax: 0532 506868



Government
Orders
Accepted
All prices exclude
VAT.

Hints & Tips

• **!Draw** – I used to find it difficult to get to grips with !Draw because I couldn't visualise the final layout. However, if you use the 'new view' option together with 'zoom' you can get a WYSIWYG view. Not only this, but it is possible to manipulate the major view from the minor view. L Kennedy.

• **Acorn DTP** – This may seem obvious but I only just figured it out! When printing documents from Acorn DTP, I found that footers (in particular) wouldn't print. I spent ages looking through the DTP manual trying to figure out why. I suddenly occurred to me that it might be the printer driver and sure enough, it was. The page size for A4 paper has margins set all round, in particular a bottom margin of 20 mm is set, and it is this which is 'clipping' out the footers. Set this to 0 (say) and the problem is solved. Kevin Quinn

• **BASIC programs from !Edit** – When in !Edit you can actually run BASIC programs etc from within the desktop without pressing <f12>. I happened to come across this accidentally. Press <menu> on the Edit icon and click on New Task Window. This will produce a task window on the screen and will allow you to execute the BASIC environment without leaving the desktop and allow you to format disks also allowing you to nearly multi-task formatting disks, you can suspend the task get on with something then resume the task and the formatter will continue where it left off! Pressing <menu> from within the task window will give a list of commands (anyone know what link/unlink does??) Graham Bisset

• **Changing your batteries** and keeping your CMOS RAM settings intact. If you connect a 100 microfarads capacitor in series with a 100 ohm resistor and clip them across the battery connections, (+ve of the capacitor to +ve of the battery) you will have about 70-100 seconds to change your batteries without losing your CMOS settings. The computer should be kept switched off throughout the operation. J W E Jones

• **Current directory path** – With reference to Richard Skemp's query in Archive 3.5 p43 about getting the full pathname of the current directory, the 'easiest on your fingers' method uses system

variables. It was difficult to implement and required a fair amount of digging in the manuals but it does demonstrate the use of *Set, *SetMacro and *SetEval, so here it is:

Place the following two lines into an Obey file, either via *Build or by using !Edit. Save it in the library directory of your current disc as "Path" and set the filetype to be an Obey file (*SetType %Path Obey). Execute it with *Path. Subsequent use of *Path will have other effects – see later.

```
Set Path$ $
SetMacro Alias$Path Dir %%0||mIf
  "%%0" = "" Then Set Path$ $
  Else If "%%0"LEFT1 = "$" Then
    Set Path$ %%0 Else SetEval
    Path$ Path$ + ".%%0"
```

How it works: "Set Path\$ \$" sets up a system variable called "Path\$" and assigns to it the string value "\$", which you will recognise as being the name of the root directory.

"SetMacro Alias\$Path ..." creates a macro system variable called "Path".

When "*Path [<directory path>]" is executed, the macro first selects the directory specified in <directory path> by way of the old *Dir command.

If no directory path is specified, the old *Dir selects the root directory. The macro examines <directory path> and, if it is a null string (""), it sets Path\$ to "\$".

If the leftmost character of <directory path> is "\$", the macro replaces Path\$ with <directory path>.

If <directory path> is not null and does not have "\$" as its leftmost character, the macro appends a full stop and the string specified in <directory path> to whatever Path\$ contains.

Path\$ thus always contains the full pathname of the currently selected directory (CSD).

You can actually redefine *Dir itself to do this, so that you don't have to remember to use *Path instead of *Dir. In other words, it becomes transparent.

All you do is to change the first occurrence of Path (in "Alias\$Path") to Dir (i.e. "Alias\$Dir") and change the "Dir %%0" to "%Dir %%0". The extra

% sign in front of Dir is to enable the original version of Dir to be used (FORTH programmers will recognise this as smudging). I have not actually tried this, but it should work!!

How to use it:

Use "Path" wherever you would use "Dir". In fact, do not use "Dir" at all (more on this later). Also use full pathnames, as opposed to wildcarded pathnames. This is not really necessary, but it looks neater.

The command "***Path Basic.Games.Adventure**", executed from the root directory, would set sub-directory "Adventure" as the CSD, exactly in the same way as "***Dir Basic.Games.Adventure**" would do.

Notes:

1. No leading asterisks. You may enter them, but they are superfluous, as these lines are sent direct to the CLI.
2. The use of two double bar characters in | | m. This is so that Obey will insert it in the macro variable as lm, instead of as ASCII 13.

Alias\$Path is a multi-line (well, a 2-line) command. This is quite acceptable, provided each line is separated by a carriage return (ASCII 13, or lm).

Note that the (*)Dir command is at the start of the macro. Putting it at the end causes errors related to the (*)If...Then...Else construct.

3. Double percentage character in %%0. This is so that Obey will insert it in the macro variable as %0, instead of attempting to evaluate a (non-existent) parameter when the command *SetMacro is executed.
4. Double quote marks where the parameter is to be evaluated as a string. This is so that string evaluation will proceed correctly.
5. Use of upper and lower case. In general they are interchangeable at will.

(*)Eval supports a number of operators (see p.394 in the Archimedes User Guide). Some are in the form of ordinary words, such as LEFT, MOD, AND, OR, etc, and others are symbols, such as *, +, >>, etc. The ordinary words must be in UPPER CASE, otherwise they are not recognised.

Disc-related commands seem to be quite happy when offered system variables as pathnames. If an application sets up its own pathname as, say, "Application\$Dir", you can make the first line of a BASIC program.

```
10 REM > <Application$Dir>.Program
```

and be confident that the BASIC command SAVE will put it in the right place, as the Filer (I think) evaluates <Application\$Dir> correctly. Also *Eval Application\$Dir (without the <>) also returns the value of the variable.

You can get the value of the system variable into a BASIC variable with the following FN:

```
buffer_length%=255
DIM buffer% buffer_length%
PRINT FNevalOS("Application$Dir")
      : REM NB!! no < >
END

DEF FNevalOS(A$) LOCAL T%,L%
SYS "OS_EvaluateExpression",
  A$,buffer%,buffer_length% TO
  ,T%,L%
IF T%=0 THEN = L% : REM integer
      result
buffer%?L%=13
=$buffer%      : REM string result
```

Anton Mans, Durban

• **Double height in Fortran** – The following short FORTRAN routine can be used for the Archimedes to 'tart up' screen output. It is a routine for printing text in double height on the screen and uses OS_Word call 10 to read the system font bit pattern. A VDU23 call is used to define a pair of characters consisting of the upper and lower halves of the input character. These two characters are then printed, one above the other, giving a double-height character. It's very useful for printing headings in screen output and doesn't require messing about with special fonts (I'm going to go on to that next!).

I claim no originality for the overall method. A routine in BBC BASIC to do this was published in Beebug some years ago (I forget who the author was), but I thought it was worth publicising a FORTRAN version of it in Archive because I don't think the translation is immediately obvious and

readers may not have been aware of the technique anyway. It should be easily extendible to double width, quadruple height, etc.

Rather than just give the bare routine, I have produced a short program incorporating it, that writes out a couple of strings. Much use is made of the '\$' format to suppress CRLF when characters are being sent to the VDU driver. The routine is more complex than the BASIC equivalent for two reasons. Firstly, although there is a word 'LEN' in FORTRAN, it only seems to give the maximum length of any given string, i.e. the length that it is originally declared to be (e.g. CHARACTER*78 TXT; LEN(TXT) gives 78). So in SUBROUTINE DOUBLE, there are a few lines to find the length of the string passed to it by counting backwards from the maximum length until a non-space character is reached. N.B. this will fail if the string is put into the argument list, e.g. CALL DOUBLE(10,20,'Hi there'). Secondly, since individual bytes in memory are not accessible to FORTRAN, you have to get the result back as three (4-byte) integers, and use IAND with ISHFT to gain access to the bytes.

```

PROGRAM DHTEST
C *****
C  A DEMONSTRATION PROGRAM FOR
      PRINTING DOUBLE HEIGHT
C  TEXT IN ANY MODE (NOT 7) ON THE
      ACORN ARCHIMEDES,
C  IN ACORNSOFT FORTRAN. USES ACORN
      EXTENSIONS: OSWORD, ISHFT, IAND
C  BY R.D.WRIGHT
C  *****
CHARACTER*78 TXT
WRITE(6, ' ($,2A) ') CHAR(22)
                                , CHAR(12)
WRITE(6, *) 'DOUBLE HEIGHT TEST'
TXT='THIS IS A VERY LONG TEST
      STRING'
CALL DOUBLE(10,20,TXT)
TXT='AND MORE'
CALL DOUBLE(10,23,TXT)
STOP 'END OF DEMO'
END
SUBROUTINE DOUBLE(ICOL,IROW,TXT)
CHARACTER*78 TXT
INTEGER BLOCK(0:2)
INTEGER B(0:8)

```

```

N=79
10 N=N-1
  IF (TXT(N:N).EQ.' ') GO TO 10
  DO 20 I=1,N
    BLOCK(0)=ICHAR(TXT(I:I))
    CALL OSWORD(10,BLOCK)
    DO 2010 J=1,3
      BLOCK(0)=ISHFT(BLOCK(0),-8)
      B(J)=IAND(BLOCK(0),?IFF)
2010 CONTINUE
      B(4)=IAND(BLOCK(1),?IFF)
      DO 2020 J=5,7
        BLOCK(1)=ISHFT(BLOCK(1),-8)
        B(J)=IAND(BLOCK(1),?IFF)
2020 CONTINUE
        B(8)=IAND(BLOCK(2),?IFF)
        WRITE(6, ' ($,10A) ') CHAR(23), CHAR(
                                240), CHAR(B(1)), CHAR(B(1))
        +, CHAR(B(2)), CHAR(B(2)), CHAR(B(3)
                                ), CHAR(B(3)),
        +CHAR(B(4)), CHAR(B(4))
        WRITE(6, ' ($,10A) ') CHAR(23), CHAR(
                                241), CHAR(B(5)), CHAR(B(5))
        +, CHAR(B(6)), CHAR(B(6)), CHAR(B(7)
                                ), CHAR(B(7))
        +, CHAR(B(8)), CHAR(B(8))
        IX=ICOL+I-1
        IYU=IROW
        IYL=IROW+1
        WRITE(6, ' ($,8A) ') CHAR(31), CHAR(
                                IX), CHAR(IYU), CHAR(240)
        +, CHAR(31), CHAR(IX), CHAR(IYL)
                                , CHAR(241)
20 CONTINUE
      RETURN
      END

```

R.D.Wright

• **Easy 'C' Compilation I** – While working on a C program it can become quite tedious to type in the same parameters every time you want to compile the latest version of your program. One way to get around this is to use something like "C-Front" from Mitre which allows you to compile directly from the desktop. A simpler and cheaper way is to use an 'Obey' file which will do the job for you. Below is an example that has been written to make the compilation of C (Release 3) programs easier. Using this method, you can still work from the desktop editing your text in !Edit, saving it, then

pressing <f12>. You can then use the following command:

```
*c <inputname> [outputname]
```

It will then automatically Compile and Link your new masterpiece!

The program has been fully commented to aid understanding. It should be saved as an 'Obey' file in the 'Library' directory under the name "C". Obviously it should be customised to suit your own directory layout and preferred compiler options.

```
| 'C' Obey file, Compiler & Linker -
| 16/02/90 Mark Taylor
| Syntax : "*C <inputname> [output
|                                     name]"
| Where 'inputname' is the name of
|       the file to be compiled
| and 'outputname' is the name of the
|       output file, e.g. !RunImage, etc.
|
| Check for parameters. If none
|       present then error and output
|       a message.
if "%0" = "" then error 0 Error in
  'C' Compiler & Linker. Syntax : *C
    |<inputname> [outputname]
|
| Set system variable 'ipname$' to
|       first parameter after the "c".
Set ipname$ %0
|
| Check for a second and if there is
|       one set 'opname$' to that
|       variable.
| else set 'opname$' equal 'ipname$'.
If "%1" > "" then Set opname$ %1 else
  Set opname$ %0
|
| Compile with 'ipname$', do NOT Link
|       (-c). ANSI 3 Standard method.
cc -c <ipname$> -I$.RISC-OSLib
|
| Link 'ipname$' with standard lib-
|       raries and output as 'opname$'.
link -o <opname$> o.<ipname$> $.RISC
  -OSLib.o.RISC-OSLib $.CLib.o.Stubs
|
| Echo finishing message.
Echo File '<ipname$>' successfully
  compiled as '<opname$>'.
| Unset the temporary variables.
```

```
Unset ipname$
Unset opname$
```

Below is a cut down version that is not quite as sophisticated but is easier to understand.

```
| 'C' Obey file, Compiler & Linker -
| 16/02/90 Mark Taylor
| Syntax : "*C name" Where 'name' is
|       the name of the file to be
|       compiled.
cc -c %0 -I$.RISC-OSLib
link -o %0 o.%0 $.RISC-OSLib.o.RISC
  -OSLib $.CLib.o.Stubs
Echo File '%0' successfully compiled.
```

• **Easy 'C' Compilation II** – This is not a new idea but it is still worth mentioning. To speed up the compilation of 'C' programs it is a good idea to copy the necessary files to a RAM disc. I have written an Obey file to achieve this. It is best saved into the Library directory so that it may be called from the star (*) prompt any time. The way the file is set up at the moment means that it will copy files from the hard disc to a RAM disc of some 1.5M. The RAM disc is set up using the "MemAlloc" module (this can be found in the !Lander directory on the Apps2 disc of RISC-OS). It should be copied in the modules directory of the !System folder. The directory layout reflects a fairly standard system under release 3 of Acorn's ANSI C.

Important!

You must remember to copy the files you have changed during your session back to disc before switching off!

```
| Obey file to set up RAM disc as the
|       working disc for 'C' 18/1/90
|
| First create large RAM disc: 1.5MB!
RMensure MemAlloc 0.11 RMload :4.$
  !System.modules.memalloc
RAMFSSize 1500
RMensure SharedClibrary 3.50 RMload
  :4.$ !System.modules.clib
RMensure FPEmulator 2.80 RMload :4.$
  !System.modules.FPE280
|
| CDir RAM:$.RISC-OSLib
| CDir RAM:$.CLib
| CDir RAM:$.User
```

```
Copy ADFS:$.RISC-OSlib.* RAM:$.RISC-
                                OSlib.* ~cr~v
Copy ADFS:$.CLib.* RAM:$.CLib.* ~cr~v
Copy ADFS:$.User.* RAM:$.User.* ~cr~v
CDir RAM:$.Library
Copy ADFS:$.Library.link
RAM:$.Library.* ~c~v
Copy ADFS:$.Library.cc
RAM:$.Library.* ~c~v
Copy ADFS:$.Library.squeeze
RAM:$.Library.* ~c~v
| Copy my single letter Obey file
'C'.
Copy ADFS:$.Library.c RAM:$.Library
                                * ~c~v
|
|Define function keys.
k.1c
k.2cc -c
k.3-!$.RISC-OSLib
k.4link -o
k.5$.RISC-OSLib.o.RISC-OSLib
k.6$.CLib.o.Stubs
k.7Squeeze -v
| Delete all unwanted object files!
k.12wipe o.* ~c
|Perform *RAM at the end!
RAM
|Select $.User directory!
Dir $.User
Echo RAM Disc now setup correctly
                                (DIR=$.User)
Echo
free
Filer_OpenDir RAM:$.User.c
```

• **Music Maestro please!** – The !Maestro application provided by Acorn is very powerful, if very complex. However, it still has a number of shortcomings relative to the old Island Logic's "Music System" for the BBC micro. These include the inability to delete whole bars or several bars; the inability to handle triplets; the inability to change Tempo within a passage. Also, entering scores by dragging icons is extremely slow and laborious and finally, "Music System" counted notes in a bar and, when the right time had been used, automatically entered a bar-line. This not only saved an operation, it also provided a cross-check on typing accuracy: the unexpected appearance of a bar-line indicated a mistake somewhere.

!Maestro's !RunImage file has been modified to permit changing of the Tempo during a passage. This required a new sprite, placed in the space in the top right corner, next to the hemi-demi-semi-quaver rest; 'M' for Metronome seemed to be a reasonable choice. This sprite can be picked up, just like the others, and placed above the appropriate bar-line. When it is 'clicked' to fix it in position, it changes into the crotchet symbol followed by " $=\langle \text{num} \rangle$ ", where $\langle \text{num} \rangle$ is a number from 40 to 210, representing the Tempo as selected by the Tempo sub-menu from the main Maestro menu. The numbers (of beats per minute) are given on page 1810 of the Programmer's Reference Manual.

During these investigations, a few small errors were noticed in the Programmer's Reference Manual (RISC-OS Edition). In Vol.4, under Music Files:

p.1809 "Maestro" is followed by "Line-feed", &0A (not &0D)

Music Data: Not "4*8 length of queue of notes..." but "5*8" because the length is written to file as an integer variable, FINE%(C%) (as in DC al fine!), so it consists of one byte (&40:integer) followed by 4 bytes (value).

p.1811 Attributes: Clef Bit 5 is unused Bits 6 & 7 are (Stave-1), range 0-3. The attributes as defined in !Maestro. !RunImage lines 28900-29020 agree with those obtained by a *DUMP of a Music file. Alan Quayle

- **Shift Caps Lock** will give you capitals normally but will enable lower case characters to be typed by holding down the shift key – very useful when writing BASIC programs. Fred Lee

- **SID** – There is a mail-server on the SID database and so if you can use E-MAIL (e.g. if you are a student at University, like me) then you can mail it at SIDInfo@uk.co.acorn with a message containing the word 'help' and it will mail you a file telling you how to download stuff. It only has the free PD stuff though (obviously). The advantages of this service work both ways – the SID machine can schedule the replies in non-peak times and you are not using lots of on-line time down-loading. Kevin F. Quinn

- **TinyDirs** – it may be obvious to most of you but... The TinyDirs program not only allows

directories to be installed on the icon bar but allows applications to be as well. After all they are only directories really. The application can then be run by clicking on the icon on the icon bar in just the same way as any other application. This means you can have your favourite programs on the icon bar without having to worry about how to install them there. The TinyDirs program will also cope with more than one directory. (A friend was running a new copy of TinyDirs each time he wanted a new directory installed.) If you just drag the icon for the directory you want installed onto one of those already there it will be added to the list. Andrew Ling.

- **Turkish 'i'** – There is a minute error in the character set supplied as "Latin3" on the Archimedes which, according to Acorn's manual, corresponds to ISO 8859/3 covering German, Spanish, Turkish etc. Character 185 is supposed to be the Turkish lower case 'i', i.e. like the English 'i' but without the dot. (*The Mac has let me down for once – no Turkish 'i'!* Ed.) Instead, it replicates the standard Latin 'i'.

To overcome this, insert the following in a boot file:

```
*echo <23><185><0><0><56><24><24>  
      <24><60><0>
```

John Morgan

- **Working on a 1Mbyte machine** – When loading applications within the desktop especially on a 1Mb machine, always call up the task display and reduce the size of the next slot to as low as the machine will allow, some applications will allow you to set it to 8k and they will just grab as much memory as they need, however some will report 'App need XXXk to start' so change the slot to about XXX+20k for workspace. If you do this then the application can only grab as much memory as it needs. However, if you leave say 640k in the next and you click on an application then its likely that the application will grab about 600k or so leaving precious little for other programs, data etc Users of 2Mb+ machines don't have the problem, but they still need to be careful of memory grabbing applications! Graham Bisset **A**

Home & Business Programs from Apricote Studios

The Account Book: Comprehensive small business accounts to trial balance. VAT approved. Absolutely the easiest program to use, with neat final books and hundreds of reports. No entry limits. "The Account Book gets first prize for both price and performance." – comparison of different business programs in Micro User, July 1989. "A truly user-friendly program. If you buy these packages, you will not be disappointed" – Beebug October 1988 & December 1989. **£27.95**

The Invoice Program: Link with The Account Book or use independently. 700 customer database, Invoices, Statements, Stock presets, Debtor lists, Mail shot labels and load more. **£27.95**

Purchase The Account Book and The Invoice Program together for **£49.95**

Personal Accounts: Just released. Multiple bank accounts etc, Direct debits. The best report, editing and search facilities. **£14.95**



Apricote Studios

2 Purls Bridge Farm, Manea,
Cambs, PE15 0ND.



Tel: 035-478-432 for information, help or to order.



Aleph One Limited

ARM3 UPGRADE



Aleph One Limited

The Aleph One Limited ARM3 Upgrade propels the Acorn Archimedes range to give a phenomenal 10-12 MIPS (Million Instructions per Second). Acorn Computers' own VLSI Design Team has engineered this latest ARM processor to give performance on a par with top class workstations - based on the same ARM2 processor core already used in the Archimedes, ARM3 is machine-code compatible yet typically runs THREE times faster! Here's what some ARM3 users and reviewers have had to say...

"Really, it is not a question of whether you can afford to upgrade, you should ask yourself if you can afford not to!"

Paul Fellows, A&B Computing.

"Time is money. You value your time, you buy the product."

Roger Howarth, BBC Acorn User.

"Our 1024-point fast Fourier transforms used to take about 150ms; now we can do them in 50ms, so we see them in real time."

A user in the MoD.

"I am writing to congratulate you on your excellent ARM3 Upgrade..."

Dr. Colin Dean, Cambridge.

"The overall speed improvement was no less than 12.6 times - a very significant increase - and produced π to 300 terms in 1 second."

Mr G H Foot, Oxted.

"...an Archimedes fitted with an ARM3 is still an inexpensive way to buy the 15 to 20 MIPS it offers on many programs."

Dick Pountain, BYTE Magazine.

When running the C Dhrystone benchmark from the Desktop in mode 12 the ARM3 cache improves the performance from 4630 to 15480 dhrystones per second, a ratio of 3.4. Many BASIC or machine code programs will be sped up more than this - the improvement is highly dependent on the task being done and the screen mode. For detailed information on performance please write or 'phone us.

The fine print... Two speeds of ARM3 Upgrade are available, with clock rates of 20MHz and 30MHz. A 30MHz upgrade costs £595 + VAT (RRP). The 20MHz upgrade costs £495 +VAT (RRP). The 20MHz upgrade is only sold direct from Aleph One Ltd. - please call for availability. The Upgrade can be fitted to A310 & A440 models with the addition of a MEMC1a upgrade, and directly to A410, A420, A440/1 and R140 models. Available from all good dealers or directly from us.

For details contact:

Aleph One Ltd. The Old Courthouse, Bottisham, CAMBRIDGE, CB5 9BA.
Telephone (0223) 811679 Facsimile (0223) 812713

MORE NEW RELEASES FROM DABS PRESS

Z88 PipeDream: A Dabhand Guide

from John Allen - expert Z88 journalist

PipeDream, the built in word processor, spreadsheet and database built into the Z88 is an extremely powerful tool, and John Allen, author of many articles of the Z88, and experienced user has produced what must be the ultimate tutorial and reference work on the software. With many example documents and templates, this book takes you from first principles to a thorough understanding of the program. Linking to PipeDream on other machines such as the BBC and Archimedes is also covered.

Only £14.95 all inclusive

AVAILABLE 24th APRIL

Also still available is our first book on the famous Cambridge portable marvel, Z88: A Dabhand Guide, written by Trinity Concepts, authors of the Z88 operating system. Also priced at £14.95

THE LATEST DABHAND GUIDE

We have moved! Please note our new address and telephone number for all correspondence.

**DABS PRESS (AR5)
22 WARWICK STREET
PRESTWICH
MANCHESTER
M25 7HN
TEL: 061-773 8632
FAX: 061-773 8290**

For personal callers Warwick St. is in the centre of Prestwich Village, just off Bury New Road.

Dabs Press books and software are available from all good bookshops and Acorn dealers, or in case of difficulty direct from the above address. Access/Visa/official orders accepted. Free catalogues on request. Prices include 15% VAT (on software) and UK postage. Foreign deduct VAT but add £2.50 Europe/surface, £12 airmail.

061-773 8632

CAMBRIDGE PASCAL

Full featured Pascal Compiler for the Acorn
Archimedes



**DABS
PRESS**

Cambridge Pascal is a comprehensive version of the Pascal language and includes many extra enhancing features over the standard specification including full dynamic string handling, local error handling, random access files, direct access to 'star' commands and RISC OS SWI calls. It is 100% Desktop compatible but can

also be run from the command line. All programs compile into fast, efficient machine code.

Cambridge Pascal requires any Archimedes running RISC OS with at least 1Mb of RAM. Hard discs and extra memory are useful, but the system is perfectly workable with this minimum configuration.

Price £79.95 (£69.52+VAT). Site licences available. Free demonstration disc and full specification sheet available on request.

DEMO DISC FREE ON REQUEST

ARCHIMEDES FIRST STEPS

Beginners Guide to the
Archimedes & BBC A3000

This book is the perfect introductory guide to the Archimedes, to guide you through the first few months of ownership, acting as an easy-to-read supplement to the User Guide.

The book describes in detail how to put the RISC OS Desktop to best use, and also documents the programs on the Applications Discs provided with the machine.

But the book also goes beyond this and describes the sort of software and hardware additions available to the Archimedes owner, and how to choose and install them.

The many features of this book include:

- Applicable to all Archimedes with RISC OS
- Using the Desktop, RAMdisc and ADFS
- Edit, Paint, Draw and Maestro
- The Task Manager
- The BBC and PC emulators
- Hardware and software additions
- Illustrated throughout

AVAILABLE NOW!

PRICE £9.95

ONE OF OUR BEST SELLERS

Comment Column

• **Caverns** – My first impressions of “caverns” are that it is great. A full review will take some time, even with the tips provided by Minerva. The game is very large and complex. Their claim that the game requires a lot of skill and patience is certainly no exaggeration. I will write a proper review as soon as possible. Philip Green

• **Confusion** by Cambridge International Software. “When I ordered this game I had expected it to be Fireball II as reviewed in Archive 3.5 p 41. This simply is not the case.

Confusion is credited to David Kelly (rather than Simon Heather who wrote Fireball II) and starts by offering you the choice of sound or not. When you choose ‘Y’, the Archimedes makes some very loud noises. If you reboot the disc and selected the ‘N’ option, you still get the sound, albeit somewhat softer.

Once the ball was fired, a strange voice-like noise is sounded and the ball proceeds to bounce around the screen. The bricks appeared to be ‘soft’ as the ball actually goes some way into them before it bounces off. I found this somewhat off-putting at first as the ball returns other than would normally be expected.

Every time you lose a life, the game plays a sampled voice that announces ‘prepare to die’ – this very soon loses its charm! Once you have lost all your lives and selected the new game option, bricks will appear in places where they should not and striking them with the ball does nothing.” A C Eggleton

Sorry for the confusion(!). Simon told us his Fireball II would be sent out through Cambridge International Software as Confusion, so we assumed that this was that same thing that David Bilsby reviewed in Archive 3.5 p 41. In fact, Fireball II is available as... Fireball II from C.I.S. at @£24.95 (£23 through Archive). Those who bought Confusion under the misapprehension that it was Fireball II can return it with £13 and upgrade to the real Fireball II.

• **Hawk V9 Colour Digitiser and FastGrab Software** (N.B. This is based on Wild Vision’s own description. However, we have seen one of these boards and were impressed, so we thought readers

might like to be informed about this new product in some detail. We hope, next month, to have a user’s view. If any readers are already using one of these boards, please let us know if they are as good as they seem. Ed.) – The Hawk V9 is a real time Colour Video digitiser designed to capture high quality colour sprites for use in RISC-OS applications. It operates under the multi-tasking desktop environment and gives facilities to produce both live colour displays on the desktop and high quality sprites which can be dragged into other applications.

The product consists of a double width expansion card (podule), which contains all the digitiser hardware, and a RISC-OS application !FastGrab which provides software for the capture and processing of images.

The digitiser accepts any composite PAL video input signal via a BNC connector on the back panel. Suitable signals can be found from almost any video camera or recorder. The signal is converted to RGB and digitised in real time as three four-bit-per-pixel data streams. These can then be stored in the digitiser’s onboard memory at a resolution of 512 x 256 pixels. The brightness, contrast and saturation of the digitised image can be controlled via software to produce the best possible image.

The digitiser is set up to produce images with square pixels – something which is most important if subsequent processing is not to distort the image. However, hardware facilities do allow the pixel aspect ratio to be changed.

Data can be read from the digitiser in either of two formats:

- i) 12 bit per pixel (4 bits each of R, G and B) – This can then be processed to provide high quality images.
- ii) 8 bit per pixel format, compatible with the desktop palette. In this mode, hardware processing is used to provide data which can be written directly to screen in 256 colour modes to give the optimum picture quality. The use of hardware processing means that a reasonable quality, “live” colour display can be produced on screen.

The desktop application !FastGrab allows several functions to be performed:

- The display of a live image on screen in a window at different sizes.
- The capture of an image and its display in a window - 8 bit per pixel.
- The processing of 12 bit image data to give high quality 8 bit sprites using a dithering technique.

The Image processing uses the Change FSI algorithm developed by Roger Wilson at Acorn Computers. This provides several features:

- The ability to produce a high quality 8 bit sprite in any mode.
- Image scaling and flipping.
- Image processing such as contrast enhancement, histogram equalisation, sharpening and edge detection.
- Conversion to monochrome images for low colour (e.g. sixteen colour) modes.

FastGrab also allows the processed images to be displayed in windows, saved to disc or dragged into other applications.

Another feature is the control of the PAL decoder hardware from the desktop. Simply dragging slider bars with the mouse gives control of brightness, contrast and saturation.

Work with images is intensive of both memory and processor power and FastGrab is no exception. For best results, a machine with at least 2 Mbytes of memory and a hard disc is recommended. However, the software has been specially designed so that it will run in less memory by saving temporary files to disc. In practice, this means that FastGrab will run on any Archimedes with 1Mb of memory, although it may perform more slowly. The software automatically senses the amount of memory available and uses disc cacheing if required. The changeover is transparent to the user. (£339 +VAT from Wild Vision or £360 inclusive through Archive.)

• **Home Accounts** - If you are having problems with any of the remaining bugs in this program, note that the latest version is 1.08. It may be that they have been fixed in this latest version.

• **Incompatibility** - I tried to use the Watford ram upgrade in my A3000 with a 5.25" disc interface

from CJE Micros. They caused all sorts of problems because they touched each other. I was advised by Watford Electronics to hold the two apart with some sort of insulating material. Even with that, it gave problems so I had to take the interface out. The moral is: before you buy a new add-on, check with the supplier if it is compatible with any the other bits and pieces you have already got in your computer. Greg Goodrum, Billericay.

• **Lack of a decent database** (This is representative of a number of letters I have been getting on the same subject.) Where can I find a decent database program for the Archimedes at a sensible price? Most of the current offerings are either IBM PC copies or they were written some time ago for the Arthur operating system, or they are ridiculously expensive:

Flying Start II (Mitre) - IBM PC clone.

Alphabase (Clares) - Written for Arthur OS.

System Delta-Plus II (Minerva) - Not RISC-OS or Multi Tasking.

MultiStore (Minerva) - Very good but £300 !!

Data Vision (Silicon Vision) - Not multi-tasking.

The alternative is to use an integrated package like Pipedream. This may be a very good package but its database facilities are its poorest part and why should I have to buy a program like that and then only use one third of it? It really can't be too much to ask for a good RISC-OS compatible database written in, for example, the same style as First Word Plus. I have spoken to Acorn and they have no intention of producing one, not even in their forthcoming Productivity Tools range. So come on Minerva or Computer Concepts or any body else, please give us a database that makes full use of the Archimedes and the desktop at a reasonable price (i.e. £140). Mark Taylor

• **MultiStore** - A letter from Nova Fisher, proprietor of Minerva Software, confirms the statements in Graham Hobson's review article last month that they have no intention of making MultiStore a programmable database. "Should there be sufficient demand for a programmable database then we will consider it. However, our present workload is such that we would not even consider a release of a programmer's toolkit version

until at least 1991." She says that if people want a programmable database, they should use System Delta Plus.

One other important point that Nova makes is that, contrary to what Graham says, MultiStore IS fully relational. Does anyone have a definition of what is or is not relational?

• **New version of RISC-OS?** – Acorn say that, despite rumours to the contrary, there will be **no changes to the RISC-OS ROM set in 1990**. Whilst RISC-OS is under constant revision, they are not intending to produce an updated version this year. In the meantime, any modifications will be supplied on disc as they did with their RISC-OS Extras Disc (Shareware 17). They also confirmed that, as and when a new version appears, A400/1 and A3000 owners will be able to fit them directly but A300 series owners would have to use one of the ram upgrade boards that have ROM sockets on them. (No mention of the old A440's, but presumably they would take the larger ROM's.)

As I have said before, I doubt whether these newer ROM's will ever see the light of day. It would not surprise me if Acorn launched an ARM3 + 8 Mbyte version of the Archimedes during the course of 1990. That's just uninformed guess-work. I don't have any inside info – they are obviously quite tight about such things! Ed.

• **Trivial Pursuit** – Some niggles that emerge after using Trivial Pursuit for a while are:

- 1) The pointer keeps getting relocated, so you have to look and see where it is before you can move it to where you want to go. This also results in having to lift and move the mouse repeatedly as it would otherwise drop off the edge of your desk.
- 2) Some of the questions are out of date – e.g.: Where will the 1988 Olympics be held?!
- 3) After playing the game, and quitting nicely, you have to reset the computer before using some other applications. Notably, the PC emulator will not start due to the font manager being in use.

I have no idea how many questions there are supplied with Trivial Pursuit; ArcTrivia had an awful lot and you could edit the question files to your liking (this was certainly necessary to get rid of the spelling errors) and to renew the question base.

(Unfortunately, I can no longer use ArcTrivia since I have my SCSI disc. Even if I unplug the podule and try to start up from floppy, the software re-initialises the podule and repeatedly re-configures the machine. I have stopped it after the 3rd re-configuration and found the SCSI icon back on the bar along with 2 floppy drives!!)

Not exactly a niggle, but I can't imagine why it was put in the game: once you have decided who is to play first, second, third etc and have entered the names in that order, the computer mixes them up randomly as soon as you tell it to start the game. Philip Green **A**

!works

!works tools #1

£10

Essential Archimedes programming tools.

Enables users' applications to output DXF files compatible with !Draw. The DXF (Data Interchange) format is used for input and/or output by several popular vector graphics packages including !Draw, AutoSketch, Euclid, LinCAD, WorraCAD and PDT.

DXF files can be read by humans, merged and edited in !Edit and ported to and from other computers.

!works tools 1 contains an explanation of a version of the DXF file format as understood by !Draw with lots of examples, BASIC DXF library PROCs and model programs. Essential for Archimedes programming in any language.

Incorporate your own vector graphics in DTP!

Jim Markland
4 Shalford Close
Cirencster
Gloucester
GL7 1WG

RISC-OS only. Needs 1MB or more.

TECHSOFT

DESIGNER intro

POSSIBLY THE ONLY DRAUGHTING SYSTEM YOU WILL EVER NEED

Created specifically for the Archimedes, **DESIGNER INTRO** is based on its big brother **DESIGNER (ARCHIMEDES)**. It shares the same uniquely logical menu structure and many of the powerful draughting routines. Enough power in fact to cope with all of most peoples' draughting requirements. Many of **INTRO's** features are shown below. There is one feature however that we cannot show - its most important one - the simplicity of use that guarantees a short learning curve and speedy drawing generation.

FEATURES

Fully menu driven with simple prompts and instructions for all routines.

User definable grid with grid lock. Hunt and lock onto nearest end of line, arc or centre of a arc/circle with a single button press.

True arcs, tangents and normals, essentials for most drawing applications but particularly important for engineering applications. Nine in-built arc types alone.

Ellipses, full or part.

Powerful delete facilities, e.g., last item, line, part line, arc, part arc, etc. (The software automatically finds intersections.)

Unique **MODIFY** command allows drawing to be redrawn a line at a time forwards or backwards. The drawing may be 'entered' at any point thus giving total flexibility to edit and alter drawings.

Automatic or manual dimensioning, user definable arrow heads/leader lines.

Text entry at any size, angle or slope. (Text output limited only by the plotter available.)

Hatching at any angle/spacing. Automatically finds enclosed boundary plus single or multiple islands.

Powerful **ELEMENT** (layer) structure.

No practical limit to the number of drawing elements that may be created.

Comprehensive transformation commands, **MOVE**, **ROTATE**, **MIRROR IMAGE**, **ALTER SIZE**.

No practical limit to the size of drawing created - limited only by disk space.

Libraries of drawings can easily be created. Library items can then be incorporated into the current drawing at any size, angle, position. Zoom, scale and pan facilities (plot-outs may be taken at any time).

Output to wide range of line plotters from A5 to A0, e.g., Graphtec, Hewlett Packard, Hitachi, Plotmate, Roland, etc. Plus screen dumps for FX/MX/RX compatible printers.

COST EFFECTIVE CAD FOR YOUR ARCHIMEDES

In creating **INTRO** TechSoft have kept the features that made **DESIGNER (BBC)** so popular. There is however one feature that we have cut back on - the cost. **INTRO** is not a toy (although you will enjoy playing with it), it is a true working system at a realistic price.

P.S. If your draughting requirements are more demanding than normal take a look at **DESIGNER (ARCHIMEDES)**, more than a match for any drawing.

PERIPHERALS

Please remember that TechSoft can supply a wide range of CAD related peripherals and accessories, e.g., plotters, plotter pens, etc.

ORDERING

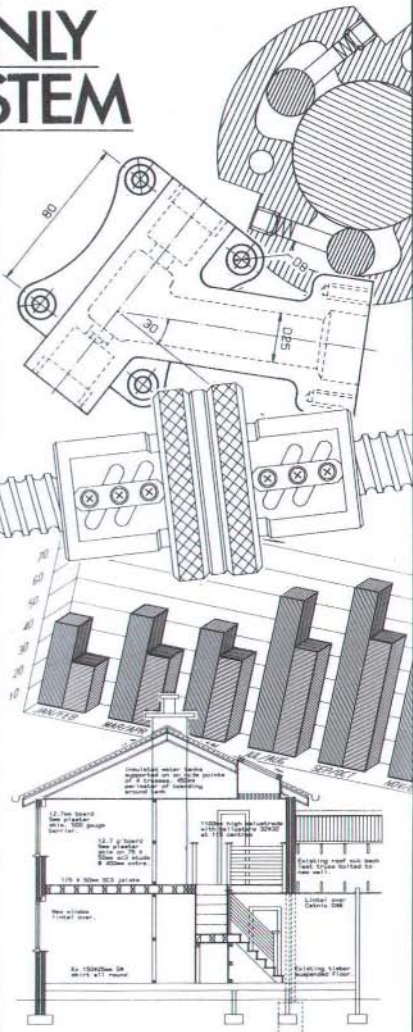
Cheque with order please, payable to TechSoft UK Ltd.
Add VAT @ 15% of total. (Official educational orders welcome.)



ONLY **£75** + VAT.
(OVERSEAS **£95**)

TECHSOFT
NEW IDEAS IN SOFTWARE

TechSoft UK Ltd.
Old School Lane
Erryngs, MOLD
Chwyd CH7 4DA
Tel: 082 43 3116



COST EFFECTIVE CAD

Help Needed and Offered

- **Disk Controller** – I am trying to write an Amiga disk reader and ADFS_DiscOp doesn't enable me to specify inter-sector gap sizes, for example. So, can anyone tell me how the VL1772 disk controller is mapped onto the memory map, please. I want to be able to program it directly. Kevin Quinn
- **Simtron** – Has anyone seen or heard anything of Simtron lately? For the last eight weeks we have been leaving messages on their ansaphone and getting no response. Does anyone know if they are still in business? Some people have sent money and had no response. Is there anyone living in the East Grinstead area who could pop in and see them?
- **VIDC registers** – Does anyone know if there is a way of finding out the default settings for the VIDC registers in each of the modes. I've read through the list of SWIs in the PRM and do not seem to be able to find anything. Andrew Ling

Help offered

- **DFS reader for Opus DDOS** – Alan Dawes has modified the DFS reader off Shareware 2 to work with multiple volumes on an Opus DDOS double density disk. If you want a copy, send Alan a disk at 102 South Park Road, Ilford, Essex, IG1 1SZ.
- **Synergy Software's Sharemaster** has been ported to the Archimedes by D G Carton. If anyone is interested to find out how he did it, you should contact him at 14 Handcross Grove, Green Lane, Coventry, CV3 6DZ.

The following helps offered's are discs which, because of their size and/or speciality, did not get on the Careware/Shareware discs. If people are interested, they should contact the authors at the addresses indicated.

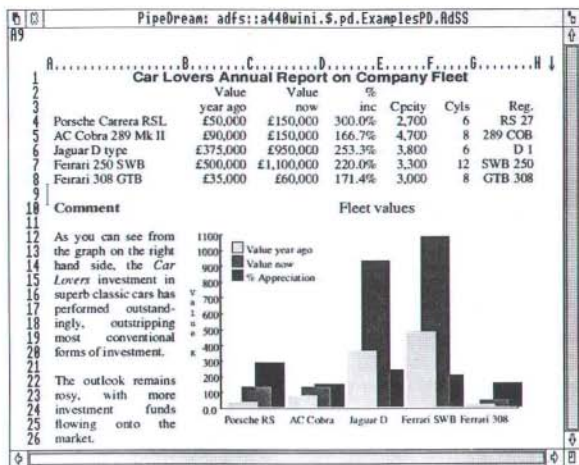
- **Tutorials covering pneumatics** for Auto Sketch and the chemical elements have been produced using a combination of First Word Plus, System Delta Plus and Gamma Plot. If anyone is interested they should contact Mr Gorringe at 51 Yew Tree Close, Yeovil, Somerset, BA20 2PB.
- **A student registration program** has been written by Ian Lynch. If anyone is interested they should contact him at 1 Melford, Off Buckingham Road, Tamworth, Staffordshire, B79 7UX.

- **Irregular outlines** – If you wish to enter irregular outlines e.g. a map Peter Trigg has written a program called 'Proput' which may help. Contact him at 11 Thurlow Terrace, London, NW5 4JB.
- **A printer circuit board designer** program has been converted from the BBC to run under the emulator. If anyone wants to know how it was done, they should contact K B Hodge at Bryn Hyfryd, 16 Mold Road, Mynydd Isa, Clwyd, CH7 6TD.
- **ROM podules utilities** to help you make the most from your Computer Concepts ROM/RAM board have been written by Compare Roberto. If you are interested, contact him at Casella Postale No 10, 18010 Badalucco (IM), Italy. **A**

Small Ads

- **A310 base + RISC-OS + manuals**, Zarch. Must go. Best offer over £550 secures. Buyer collects. Phone Kevin on 061-962-6900 evenings.
- **A310 base + RISC-OS**, with backplane, Acorn 20M hard disc and podule £890. Phone Andy on 01-733 3590.
- **A310 base + RISC-OS**. Extra software & manuals £600. Phone 0271-850355.
- **A310's** – several available, some with monitors, RISC-OS manuals etc. Prices from £550. Ring 0272-342180 (9-5.30).
- **A440 base + RISC-OS**, 40Mb hard disk, CC ROM board (with spell check) and software – Acorn DTP, PipeDream, TimeWatch, DiscTree and many others. £1250 o.n.o. Phone Simon Coulthurst on 0532-842780.
- **E-Type** £12, E-Type designer £10, Corruption £18, Holed Out £10, Pacmania £12, Grey dumps £7, Arthur PRM (still useful) £20. All in good condition. Telephone Michael Pargeter on (0462) 434061.
- **Philips CM8833** stereo colour monitor, boxed, with manual, perfect condition £170. Telephone Alan Phillips on 0942-717454.
- **Software Developer's Toolbox** £90. Call Richard Simpson on 0252-373809. **A**

PIPEDREAM 3



PipeDream 3 breaks down the barriers between word processor, spreadsheet and database. You can include numerical tables in your letters and reports, add paragraphs to your spreadsheets, and perform calculations within your databases.

Based on PipeDream 2, the best-selling integrated package for the Archimedes, PipeDream 3 has been completely re-written to take full advantage of RISC OS - if you can use RISC OS, you can use PipeDream 3. It is fully multi-tasking and multi-windowing, so you can work on many documents at once and instantly move information between them. And since PipeDream 3 can automatically load and save most popular file formats, including VIEW and First Word Plus, switching to it from other programs has never been easier.

Power, flexibility, speed, ease of use. PipeDream 3. Breaking down the barriers.

For a free brochure, see your Archimedes dealer, or phone us on 0954 211472 or return the coupon.

PipeDream 3 is for all Archimedes computers with RISC OS and 1Mbyte of RAM.

PipeDream 3 costs £147.00 +VAT.

Major features include:

- many documents loaded at once
- intuitive RISC OS user interface
- displaying and printing of pictures within text
- built-in 93,003 word spelling checker
- file compatibility with PC & Z88 PipeDream and BBC View Professional
- background recalculation
- keystroke compatibility with Z88 & PC PipeDream
- Z88 filing system
- automatic loading of VIEW, ViewSheet, Lotus, First Word Plus, Tab and CSV files
- automatic saving of VIEW, Lotus, Acorn DTP format, Tab and CSV files
- multi-field sorting
- use of all available fonts
- 62 spreadsheet functions
- external references for 3-D modelling
- macro file recorder
- slot protection

For a free brochure, complete and return this coupon

PipeDream 3 ☐ View Professional ☐

Name _____

Address _____

Post code _____

Colton Software, Broadway House, 149-151 St. Neots

Road, Hardwick, Cambridge CB3 7QJ, England.

Fax: 0954 211607 Tel: 0954 211472

All trademarks acknowledged. The chart in the screen shown above was produced by sending numbers from PipeDream 3 to Lingswally's Presenter 2 and then loading the resulting graph back into PipeDream 3.

Colton Software, Broadway House, 149-151 St. Neots Road, Hardwick, Cambridge, CB3 7QJ, England.

Fax. 0954 211607 Tel. 0954 211472

Hardware Column

Brian Cowan

I receive a fair number of letters dealing with the question of Mac discs and the whole question of emulating a Mac micro on an Archimedes. The question of discs is easily disposed of – I have discussed this before, but some readers are still not convinced! There is a fundamental incompatibility between discs from a Mac and those from other machines. Whereas most computers, including the Archimedes, spin the discs at constant angular velocity, the Mac discs operate at constant linear velocity. The rotation rate decreases as the head moves towards the centre. This means that the construction of a Mac disc drive is somewhat more complex than on other machines.

What is confusing is that some Mac computers can read and write discs in DOS format. This is possible with the drives that Macs use because they are variable speed but the converse is not true because you cannot vary the speed of the Archimedes drives so I can see no way that an Archimedes drive could cope with Mac discs. Some of you may have seen advertisements for a device for running Mac discs on a PC, but this simply consists of a Mac-type drive which is connected externally.

Initially, I was quite keen on the idea of a Mac "emulator". In fact there is such a thing for the Amiga, although it works out quite expensive since you have to buy a set of real Mac ROMs to run it. It may be (see below) that a hardware Mac emulator for the Archimedes may not be all that far away.

DOS card?

With the demise of Mach Technology, it seemed that there was no hope of a '286 DOS card appearing. Mach's price was so low that no sensible company would consider competing. However, there is some good news. I understand that there is a company working on a '286 card for the Archimedes although at present I know neither the price nor its facilities. For more information, watch this space.

Tube podule

Regular readers will know that I have been pushing the idea of a Tube podule. This would allow the use

of the old BBC second processors, including the 512 card which contains a '186 processor running DOS Plus. A tube podule should be available soon – probably supporting up to four second processors running together! The problem is that the 512 board will probably not run directly without some ARM code being specially written for it. Nevertheless, this is an exciting prospect. It seems that there may well be a 68000 second processor card produced by the same company which opens the door to Macintosh emulation.

Mass storage

In a previous issue I wrote about the 20M removable disc cartridge system from Cumana. I understand that these are now available for the Archimedes and I am making arrangements to test one out. This operates through the SCSI bus so that a SCSI interface is also required.

Also using the SCSI interface is Cumana's 500 Mbyte magneto-optical drive. As yet, an Archimedes version is not available, but it should be soon. This will be too expensive for me to purchase but as soon as I have tried one out I will let you know.

D and E format discs on old BBC's

The L format for floppy discs is the one used by the ADFS on the 6502-based BBC computers. So, by using the L format, you can swap discs between old BBC's and an Archimedes. Unfortunately one does not usually want to use L format because on the Archimedes it runs slower, as well as having a smaller capacity. The ideal solution would be to have D and E formats supported on the old BBC machines. This would require filing system ROMs to be written in 6502 code, not something I would fancy doing, but it would be useful. Any offers?

ARM3 versus FP podule

Quite a few people have asked how best to can upgrade Archimedes computers in order to increase the calculational speed of the machine. If you have a 310, the only way is an ARM3 and a MEMC1a upgrade because it doesn't have the connections on the backplane for an FP podule. With the old 440's and the new 400 series machines, you can choose between an ARM3 and a floating point coprocessor

podule. In making such a choice one must consider the following points:

- 1) The FPU will not speed up BASIC V programs at all. My understanding is that it will speed up FORTRAN programs between three and five times if they are "floating point intensive".
- 2) The ARM3 will speed up everything. On average, BASIC V programs run some four or five times faster and compiled programs have perhaps three or four times improvement, but this is irrespective of the proportion of floating point operations.
- 3) On the old 440's, both upgrades require the installation of the MEMC1a – and that alone will speed up certain types of calculation, particularly those which make heavy use of the CPU's hardware multiply command. When purchasing an ARM3 most firms will throw in a MEMC1a quite cheaply and if you buy an FP podule, you should get the MEMC1a upgrade free of charge.
- 4) The situation regarding BASIC programs is clear since they make no use of the FPU. For compiled programs, the benefit of using an ARM 3 reduces when you have large amounts of data so that the RAM cache must frequently be reloaded.
- 5) Aleph One is not the only supplier of ARM3 upgrades. There is at least one other company selling them at present: Gnome Computers are advertising

them for £495 plus VAT, and Watford Electronics will be selling an ARM3 upgrade in due course, probably even more cheaply.

6) Personally, I don't regard the FPU as a serious product. It still needs a floating point emulator to help it along as the chip does not implement everything necessary. In other words, the FP chip used (manufactured by another company) is not entirely suited to the ARM's needs. I understand that Acorn might be working on their own FPU, which would be a much more exciting product and one would also expect ARM4 to include this FPU on-chip but that is for the future!

In general then, I think the ARM3 must be the better option, although there may be certain particular applications which would run faster using the FPU. I hope these comments will be of some use to you.

(Remember what we said last month that in order to help you judge which will be best, Aleph One provide a program speed testing service. You can send them a typical program on a disc and they will run it with and without the ram cache on the ARM3 so that you can judge the likely speed increase. However, in order to help Aleph One provide that service without too much hassle on their part, they lay down certain specifications about the program to be tested. If you are interested, send them an S.A.E. for a specification sheet. Ed.) **A**

SCSI Column

Paul Beverley (who?)

Not much to report this month apart from some price changes. Oak have reduced the prices of some of their larger drives. You can check the Price List for the latest figures but one of the most significant changes is that the 70M drive has been replaced by an 80M drive at a lower price than the 70M. The external drive sizes are now the same as the internal ones: 20, 45, 80, 100 and 200 although there is also a 330M external. (If you work out the cost of drives in terms of £/Mbyte, the 80 is a very good option at £10.12/M compared with 100M at £11.30/M.)

One other effect of the price reductions relates to the option I suggested in Archive 3.4 of combining Computerware SCSI drives with Oak podules to save money if you were wanting a 100M or 200M

drive. The reduction in price of Oak's 100M and 200M drives, together with the other factors I mentioned in Archive 3.5 p 12, make this a rather less attractive option. The Archive prices are now:

	Oak pod + drive	Oak pod + C/W drive
100M internal	1130	1010
200M internal	1470	1450
100M external	1270	1110
200M external	1580	1550

A3000 SCSI podules?

Does anyone know of any manufacturers actually producing SCSI boards for A3000? As yet, there is nothing available that I am aware of other than the

Oak Computers ones which are external full-size podules in a metal case and not an internal minipodules. Mind you, they are very fast and come with a wide range of drives (20M to 330M!) and they are available through Archive. We carry 20M, 45M and 80M in stock and order the others from Oak as and when we need them.

I have removed the Lingenuty A3000 SCSI podules and drives from our price list (they first appeared in September 1989!) simply because they are still not available. We still have outstanding orders sent in by readers several months ago but we are not able to fulfil them, I'm afraid. Lingenuty have given us various good reasons why they have not been available at various stages but it is rather annoying for those who have waited patiently for such a long time. If any other SCSI minipodules

become available before those from Lingenuty, we will let the individuals know and then put the information in the next month's magazine.

(Computerware SCSI minipodules are still not available yet but Akhter Computers are threatening to have some available "at the end on March".)

Gigabyte TapeStreamer?!

Yes, Oak Computers now have, on sale, a 1 Gigabyte SCSI tapestreamer – 1,000 megabytes! It uses a tiny DAT (digital audio tape) cassette tape similar to the cassettes used in dictating machines. It has to be seen to be believed – you would not think it possible that you could get so much data onto such a small amount of magnetic tape. Mind you, it's not cheap at £3999 + VAT (£4030 inc VAT through Archive). **A**

Ultra Intelligent Machine?

Pete Holdroyd

In this review, I am not going to concentrate on the nature of the game – others have done that – but on its presentation, both on paper and on the screen.

Much has been said elsewhere about UIM's similarity to Elite, and my limited foray into it supports this view. The box it comes in is the same size, there is a "Submariner's Manual", which appears to be at least as comprehensive as Elite's, and the graphics are similar in style to those of the earlier program. Their appearance on the Archimedes was not greatly different from the Model B/Master version of UIM, and 4th Dimension will certainly win no prizes for them.

I have to admit to being something of a 'Games Tiro', so it might not surprise you to know that the enormous amount of data in the Manual was daunting. Let me explain: like Elite, it is a game which combines shoot-'em-up with trading in commodities. Unlike Elite, players can build up a portfolio of shares on a stock market but before anything really gets going, they need to acquire a 'Central Computer'.

There are 32 items of equipment, 16 weapons and systems, countless commodities and processes which can be applied to those commodities and the entire scenario has 256 levels, each of 256 locations

– called 'ports' in this game. In short, it is massive and I don't know how anyone remembers these, let alone all the keyboard operations for navigation and the 16 types of 'submarine' which occasionally flash past the screen.

Around this time I began to lose interest. It is also around this time when I could imagine a keen games player beginning to sit up and take notice. The manual strongly advises you to note down the cost of all the commodities available at each port, the port's coordinates and something called the 'operating index'.

Now, one of the reasons I enjoy using computers is because I actually dislike setting pen to paper. I am very disorganised in this respect, so I think it's a pity this information is not in the manual. Perhaps it's part of the fun.

Everyone who enjoyed Elite will enjoy UIM. So will many who never saw the earlier game but have the ability and – above all! – the TIME to fathom out the complexities of a very involved scenario. The somewhat (for an Archimedes) primitive graphics and perfunctory sound effects will probably count for little in pursuing the Great Quest. However, it is definitely not for those who normally spend their time more constructively. **A**

INVESTIGATOR 1.02

£24.95

THE SERIAL PORT

Investigator is a disc utility program which is compatible with the Archimedes 300, 400 and 3000 series using either the Risc OS or Arthur 1.2 operating systems. *Investigator* can perform the following operations on discs of many different formats:

- Examine the format of the disc in detail.
- Edit the data stored on the disc.
- Make backup copies of the disc.
- Store the entire disc as an ordinary ADFS file. This may allow archives of several master discs to be stored on a single backup disc. The disc may then be transmitted to a remote site using a modem and a communications package such as ARCTerm.
- Fully WIMP front end.

The Serial Port, Burcott Manor, Nr Wells, Somerset BA5 1NH
Voice (0749) 72234 Data (0749) 79794 (300-2400 baud)

Please send me ☐ INVESTIGATOR 1.02 at £24.95, cheque, PO enclosed.
☐ Data Sheet

Name _____

Address _____



MS-DOS Column

John Eden

In the February MS-DOS column, a few errors crept in to the short 'Grep.bat' routine as the result of the rather convoluted process by which this column reaches you. The program should have read as follows:-

```
echo off for %%i in (%2) do find
    %1 %%i
```

Also, in my description of the Drivparm command, I said that drive C: is always assigned to the first fixed disc on the system. While this might be true of a real PC, it does not apply to the PC Emulator as users with more than two floppy drives will confirm.

While I am on the subject of floppy drives, users of PC emulator version 1.33 may have noticed that it is no longer possible to have more than three floppy drives configured. If you have configured four floppies, starting the emulator results in the error message 'PC Emulator Internal Fault'. More news on this when Philip Colmer of Acorn gets back to me.

SCSI drives and the emulator

A large part of my postbag this month has been from people who are having problems running the emulator from a SCSI hard drive. Version 1.33 of the PC Emulator is compatible with the SCSI filing system and was released by Acorn to allow the emulator to be installed and run easily from SCSI drives. Within the !PC application directory there are two versions of the !Run2 file - one for use from ADFS and the other for SCSI. If you have created a DOS partition on your hard drive using the !SCSIDisk application, then the correct !Run2 file will have been installed. There are one or two occasions, however, when this may not have happened and if you are having problems, you should check this first. If the !PC application directory contains a file called !ADFSRun2 then all should be well. If, however, you find a file called !SCSIRun2 then the re-naming process failed for some reason. If this is the case you should re-name !Run2 as !ADFSRun2 and !SCSIRun2 as !Run2.

Since I don't have ready access to a SCSI drive, I would be very interested to hear from anyone who is using the PC Emulator in this way, whether you

have experienced any problems and if so, how you solved them.

!PCDir

For those of us who have to grapple regularly with Acorn's GETFILE and PUTFILE utilities, this little application from Keith Sloan comes as a godsend. Basically, it allows MS-DOS format discs to be written to and read from the desktop. The version I am reviewing here is 0.9h, although I understand a new version is to be released soon.

Starting the application installs a new drive icon on the icon bar. This will either be a floppy or a hard drive depending on how you have configured the application. A drive letter will also be shown reflecting which drive will be accessed for MS-DOS reads and writes. Users of single floppy drives are well catered for, as the application can be set up to prompt for disc swaps, allowing the same drive to be used for both ADFS and MS-DOS operations. In its present form !PCDir can only access the DOS partition (Drive_C) on drive 4, so if you have a second DOS partition on drive 5 you will not be able to access this. If you have 5.25" drives connected to your machine, you will be pleased to hear that !PCDir is able to handle these as well.

Clicking on the drive icon opens up a directory viewer for the root directory of the disc. The contents of the directory is shown as a series of rectangles each containing a filename or sub-directory. The background colour of the rectangle helps to identify this with white for a file, yellow for a sub-directory and blue for the disc label. Pointing at a file and clicking the <menu> button displays information about the file size, date stamp, etc and also gives the option to delete the file. Clicking <select> on a sub-directory closes the directory viewer you are in and then opens a directory viewer for the sub-directory selected. Because of this, it is not possible to move files around within the DOS directory structure. To move back to the parent directory you have to click on the window's close icon and when you reach the root directory the close icon has its more usual effect of closing the window altogether.

Transferring files between filing systems is simplicity itself – just ‘drag’ the file from one directory viewer to the other. !PCDir performs some file conversion processing when transferring certain file types and, for instance, text files copied from ADFS to DOS will have the end of line converted to CR LF and <ctrl-Z> will be added as an end of file marker. Text files moved in the other direction have LF’s stripped and the <ctrl-Z> removed. File names are truncated when copying to DOS and the extension given reflects what type of file it is (.TXT for text .BAS for BASIC, etc). When copying files from DOS to ADFS, the extension is used to determine what file type to set, so if you were to move a file with a .TXT extension it would be given a file type of FFF.

While it is possible to make a selection of files from an ADFS viewer and ‘drag’ them all in one go to the DOS viewer, sadly it is not possible to do this in the other direction. (*This is going to be available in !PCDir 1.0. Ed.*) !PCDir does however support direct loads and saves and it is therefore possible to save work directly from an application to a DOS disc or drag a file straight into an application for processing.

Conclusion

There are one or two extras I would like to see included in !PCDir – like being able to mark and drag a group of files from the DOS viewer and the ability to move files from one DOS directory to another (any chance Keith?). However, as it stands, I still find it an invaluable tool and would be quite lost without it. It does all it sets out to do and after using it, you won’t want to use GETFILE and PUTFILE any more. The current status of !PCDir is ‘careware’ and it is available on Careware Disc 4 which is worth the £6 for this utility alone.

CONFIG.SYS and its uses

Peter Wickes has written to me asking if I could explain what constitutes legitimate contents for a config.sys file. Well I’ll do my best. Config.sys is a special file used by DOS to alter or add to the machine’s basic configuration. The file consists of lines of plain text and is not dissimilar to a batch file. The file is read once only when DOS is booted up, so any alterations you make to the commands in the file will only take effect when you re-boot DOS.

One of the chief uses for config.sys is to enable you to load device drivers which act as an interface between the hardware device you are trying to control and the operating system. Some of the more common device drivers you are likely to come across are mouse.sys, ansi.sys, driver.sys and ramdrive.sys. As you see, all these files have a ‘.sys’ extension and this is a good indication that the file can only be loaded from within config.sys. Some software packages include their own device drivers and these can be loaded and installed in the same way.

In addition to allowing you to install device drivers, config.sys also lets you alter the default machine configuration and there are several commands you can include in the file to do this. The following are all allowed:

BREAK=[on/off] controls how often DOS checks for <ctrl-break>

BUFFERS=number sets the number of disc buffers (filing system workspace) used by DOS (range 1 to 99). Default is 2.

COUNTRY=code sets up the date and time format, and currency character to be used. Code is the international telephone dialling prefix of the country being set (i.e. 044 for the UK).

DEVICE=[drive:]path/drivename tells DOS to install the device driver ‘drivename’ (a file with .sys extension). A separate DEVICE command should be included for each driver you want to install.

DRIVPARM=/d:dd/c/f:ff/h:hh/n/s:ss/t:tt allows the disc drive parameters to be changed (see February’s MS-DOS column for details).

FCBS=x,y sets the number of files which can be open at any one time using file control blocks.

FILES=number sets the number of files which may be open at any one time (range 8 to 255). The default is 8.

LASTDRIVE=letter specifies the last drive letter DOS will recognise as valid (range a to z). The default is E.

SHELL=[drive:]path/name allows you to specify a different command processor. DOS automatically loads COMMAND.COM from the boot disc unless the config.sys file contains a SHELL command.

STACKS=number,size reserves memory for temporary program use.

As far as I know, these are the only commands that are allowed within the config.sys file. Some commands you are unlikely to use like FCBS and STACKS and most have sensible defaults anyway. The commands DRIVPARM and DEVICE may be included more than once and this allows you to make multiple changes or install several device drivers in one go. Any further tailoring of your system – like setting a prompt – should be included in the

AUTOEXEC.BAT file which is automatically run after config.sys when the system is booted.

Contact

Keep sending me your letters, ideas and suggestions. The address (as always) is: John Eden, 13 Cranleigh Gardens, Luton, Beds LU3 1LS (no phone calls please). Or, if you prefer, you can write to me care of Archive. Don't forget to include a stamped, self addressed envelope if you want a copy of the MS-DOS compatibility list. **A**

Competition Corner

Colin Singleton

Puzzles past

The January puzzle (24 square tiles) was given an overwhelming thumbs down! Pity. I thought it was an interesting problem when I first saw it. Perhaps I will stick to computer solutions in future.

The February puzzle (30 dice) has attracted some interest but, after the embarrassment of receiving what would have been the winning entry to the December puzzle after I had declared the winner, I will keep it open a while longer.

For Easter, a puzzle concerning Easter!

Easter Day is "The first Sunday after the first Full Moon on or after March 21". This was laid down by the Council of Nicaea in 325 AD. Calculating this for future years is not so easy, and is complicated by the fact that the date of a full moon depends on which part of the world you live in! Over the centuries, the Church evolved tables which are now used to calculate the official date of Easter (which may or may not correspond to astronomical fact). In 1876 an anonymous contributor to Nature gave the first correct arithmetic algorithm to replace the tables.

The algorithm consisted of ten division sums. The table opposite shows how it was presented in 1876. Integer arithmetic is used throughout. In each step a dividend is divided by a divisor giving a quotient and a remainder. 'X' is the year AD on the Gregorian Calendar. Easter is day 'p' of month 'n'.

Now it is time for you to start work. What I want to know is, which date occurs most frequently as Easter Day? The objective is to write a program to

calculate the date of Easter for the necessary range of years (when you have decided what that range is) and list the relative frequencies of the various possible dates, in the shortest possible time. You will have to code the calculation as efficiently as possible. If you use the expressions given above you will get the right answer, but will certainly not win the prize.

Entrants are asked initially to send a postcard stating the number of years you calculated, the time taken and the most frequent date. Those short-listed will be asked to submit their programs for timing on a standard machine.

Entries please, and comments, either via Paul at NCS, or to 41 St Quentin Drive, Sheffield S17 4PN.

Happy Easter!

	Dividend	Divisor	Quot.	Rem'r
1	x	19	a	
2	x	100	b	c
3	b	4	d	e
4	8b + 13	25	g	
5	19a + b - d - g + 15	30		h
6	a + 11h	319	m	
7	c	4	i	k
8	2e + 2i - k - h + m + 32	7		l
9	h - m + l + 90	25	n	
10	h - m + l + n + 19	32		p A

DTP Column

Ian Lynch

As promised, I have dedicated much of this issue's column to a review of Impression now that it is on general release. I have received a preview copy of Clare's Tempest, which looks very promising though there still seems to be quite a lot of work to do on it. Despite having telephoned and written to Beebug I have not received any information regarding Ovation.

Impression 1.03

Impression comes on 4 discs with a manual in a ring file, a copy of a textbook on DTP and the hardware key which fits onto the parallel printer port. The whole package is very comprehensive, and detailed discussion of every feature would more than fill the whole of this issue of Archive! In all honesty, I have not had time to test every single facility in every possible combination, and undocumented bugs could still exist.

To put this in perspective, I think it fair to say that a lot of software products (most?) have bugs and these are only put right with 'release 2' which often costs a good bit extra. My understanding is that CC will put right any bugs and minor functional annoyances that arise, without charge, as soon as they can. However, any major new features will be in a later release and probably only be of use in rather specialised circumstances. It will take a good number of people prolonged use to be absolutely sure no bugs exist but from this point on, bugs are likely to be fairly obscure. I will attempt to outline all the main facilities and comment on things that I have found particularly helpful or annoying. If anyone thinks I have missed anything important, please write in.

Disc one

Disc one contains the main program application and printer drivers. A directory called extensions provides facilities for loading and saving text in the following file types; Basic, ADTP, First Word Plus and Interword. Text files can simply be dragged into frames in the main window. The extensions directory also allows files to be saved in order to print in text mode (for increased speed) on Deskjet and Epson compatible (LX and LQ) printers. Some

features such as headers and footers and graphics are not catered for by this system, but most styles such as bold, underline and italic are – if your printer supports them. The disc includes the RISC-OS printer drivers which will produce their usual high quality output.

Disc two

Disc two contains several Impression documents as examples and a tutorial directory for use in conjunction with the manual. A utilities directory has an application called !Fontdraw to turn text into drawfile objects. The characters can then be rotated and scaled in !Draw before including in a document. !Newmodes provides 24 more screen modes which give a wide number of display options complimentary to those already implemented.

Finally !Supermono provides half-toning when using monochrome screen modes. This produces improved grey scales.

Disc three

Disc three contains line art and sprites as examples. These range from small mathematical symbols to some most impressive lineart and digitised pictures.

Disc four

Disc four contains the Acorn outline fonts which are licensed to Computer Concepts. It includes a Greek font which I haven't seen before and which will prove very useful in producing mathematical and scientific documentation.

Documentation

The manual is well-produced, 250 pages long, clear and comprehensive. I found little need for it because the vast majority of operations are intuitive to anyone who knows RISC-OS and a little about wordprocessing and DTP. However, to get the best out of the software the manual does help! Things such as fixed space (ALT 160) for keeping two words together instead of breaking them at a line end are in the manual and might take a long time be discovered by experimentation.

The hardware key is unobtrusive and CC will provide a short cable to let it "dangle" if it gets in the way. It only appears to affect the main program;

utilities such as !Fontdraw work without it being present. I have not found any clashes or interference with any other software or hardware I have used.

The main application

The main application is represented by a quill pen icon and is installed and run by the usual RISC-OS methods. The application consumes approximately 320k and it is quite easy to operate with a 128k font cache on a 1M machine for much of the time. There are no drawing facilities, but frame borders can be used to draw vertical and horizontal lines and !Draw files are so easy to import that the convenience to 1M users of less memory usage probably justifies the omission of these. After all, the whole point of a modular system such as RISC-OS is to minimise the need to re-invent the wheel in every application by using co-operative multi-tasking.

I have usually found that tables and line intensive pages are better done in !Draw and imported or printed directly, rather than in Impression or Acorn DTP anyway. Impression generally uses the RISC-OS menu conventions, though there are some options which use very attractive control panels. I have heard some criticism of this but I can't seriously believe that these minor departures do anything but enhance the product both in terms of presentation and functionality and are therefore, in my opinion, quite justified.

With the quill icon on the icon bar, placing the pointer over it and pressing <menu> produces the following options. *Info*, the usual information dialogue box, *new view* which gives a list of the currently available documents, *preferences* which allows the setting of certain default parameters such as document scale, automatic spellcheck load, and *units*.

Minimise memory saves non-current parts of documents to disc so that documents larger than the available memory can be processed or other applications can be loaded even when large documents are resident in Impression. Since the document on disc is acting as if it were part of that in memory, the RISC-OS filer should not be used to rename or delete a document you are currently working on. Quit will remove all documents and the application after asking for confirmation.

Clicking <select> on the Impression icon opens a window on a blank page. <Menu> produces a list of

only 5 options. A nice feature of the Impression menu structure is that you are never faced with so many options that finding the one you want is difficult. Irrelevant options simply do not appear or are grey. The first option, *document*, provides loading and saving facilities in the normal RISC-OS manner. *Info* uses the attractive "non-RISC-OS" dialogues previously mentioned to provide information about files, graphics, etc while *loading* and *saving* options are also accessed from this sub-menu. *Printing* options are comprehensive, including a pamphlet option which will allow printing of 4 pages on one sheet so as to make a pamphlet when the sheet is folded. The scale view option allows the document to be viewed at any scale size. Certain popular options such as 80% (which just fits the width of the screen in mode 12) are there at the click of a button. This last feature was not present in earlier versions and I (and perhaps others) suggested it as an improvement. This does inspire confidence in the fact that Computer Concepts actually listen and take action.

Editing facilities

The next main menu option is *Edit* which provides facilities for cutting and pasting both text and frames. These operations are intuitive and act instantly. Larger documents can be organised in chapters which are self-contained sections of the document based on a particular page style. The page style is governed by master pages which may have particular headers and footers or standard frame lay-outs.

The edit menu allows new chapters to be started, inserted or altered (by choosing a different master page for example). New pages can be forced and pages deleted, though new pages are usually inserted automatically when text is about to flow from a previous page. The final option on the Edit menu allows the standard master pages to be viewed. From here, using standard RISC-OS menus and the Impression dialogue, new master pages can be designed which can then be used in any future documents. It would be very easy to customise a style so that organisations or individuals had their own distinctive layout.

Styling

The style option allows you to highlight any section

of text by dragging the mouse and then apply certain pre-defined styles such as italic, underline, etc. What is more, each style can be edited or a new style defined by clicking on an Impression dialogue. As little or as much of each style may be altered as required, but you may even change the colour of the text and background or the font aspect ratio. A ruler which is amazingly realistic can be used for tab setting and carriage return margins. If you have used Interword, there is a familiar feel to things but the whole thing is so much quicker and better presented. A vertical ruler would be a help in some instances but since most formatting takes place in the horizontal plane this was perhaps not thought necessary. It is one feature I would like to see added in a later version for the sake of completeness.

Frames

The fourth option on the main menu allows comprehensive frame management. Frames allow text or graphics to be imported. Graphics keep their aspect ratio unless you specify otherwise which is a major advantage when importing graphics and text which need to be precisely dimensioned. Another real advantage in use is the automatic scrolling of the screen when the pointer reaches a window boundary. I find Acorn DTP very annoying in use, having to let go of the frame, scroll the screen and go back to the frame. It is a small point but one which makes a big difference to the user. The feel of frame movement is light and convenient with frame dimensions in view and automatically up-dated as the dimensions of the frame change. Frames can be transparent, opaque, have a variety of borders, repel text, be snapped to guides, etc.

The final option, misc, provides searching options, spelling check via Spellmaster, expansion of abbreviations as you type, kerning, index compilation and insertion of dates, time and page numbers.

Conclusion

Impression has been a fairly controversial product, with delays getting it finished and the illustrious dangle. All that aside, it is obvious that a lot of thought has gone into the ergonomics of both DTP and wordprocessing in Impression's design. It is difficult to convey the ease of use of a product in a text based review, but I used Acorn DTP for some time before using Impression and have found going back to it a bind, or more precisely, sheer frustration on a 1M machine. This is the acid test as far as I'm concerned.

Usually if one gets used to something first, its use tends to persist because of familiarity. One factor that may be significant is that I used to be an Interword user, so perhaps the fact that <ctrl-S> changes case and <ctrl-D> deletes a word provides a consistency which is familiar to me but I think it is more than this. A well thought out application which is written directly in machine instructions for a 4 MIP processor is a fairly unique animal – and it shows. I suppose that a full feature drawing package which combined all the text handling features of Impression, was not restricted to frame types and did everything instantly on a 1M machine would be an improvement, especially if it was made available as unprotected shareware. Then again, there is a real world out there and I certainly don't know of any software likely to fulfil all these criteria in the very near future! **A**

First Word Plus Column

Stuart Bell

Let me stick to my tradition by starting with an apology. Contrary to my review of FWP2 in January, you can unload the dictionary and hence reclaim an extra 146k or so of memory. No excuses! The manual does describe the 'End Spell Check' option – although it doesn't mention that the memory is reclaimed which is, surely, the main point of the option. How did I find this out? Did some kind person write to me? In the 'Hints and Tips' section

of a 'rival magazine', I read that 'contrary to statements made in a rival magazine...' Is this a new way of disseminating information about the Archimedes? If it catches on, the Beeb will broadcast corrections to ITN's News bulletins, and The Times will start a slanging match with The Independent...! End of soap-box, back to First Word Plus!

LQ bug-fix 1 (and 2?)

Peter Howard offers the solution to the 'sheet-feed bug' with an Epson LQ1050 printer which I report-

ed in the March column. Reading his solution, I wonder if it may also solve the 'graphics gap' problem to which an LQ850 user sought a solution. Peter recounts his tale of woe over attempts to get a bug fix from Acorn and quotes the final response from Acorn's Peter Dunn, which solved the problem for him. He quotes verbatim, as do I:

"Load the appropriate HEX printer driver from the directory \$.1wp.hex on the FWP utility disc... Insert a * before the first character in the line which is described as 'Vertical tab to line' - the line number in the hex file is 83. Resave the file with a new filename. Run the program 1stInstall (follow the instructions on page 234 of the First Word Plus User Guide)."

Obviously, the directory and page references are based on FWP1. For FWP2, the directory is \$.1WP. Print.Sources, the conversion is done by !1stConvrt on the utilities disc and the page reference is 148. I note that the manual (FWP2 p.139) says that, 'There

is a potential problem when using vertical tabs on some printers...' Since the absence of a vertical tab facility should only have a negligible effect on through-put, I hope that troubled LQ users will find that it solves their problems. Many thanks to Peter.

Panasonic KX-P1081 Driver

Stephen Mansfield has produced an 'all-singing all-dancing' driver for his KX-P1081 printer which seems to do everything that FWP2 can ask, in every pitch. I'm sending him a disc for a soft-copy and will then hold it 'in stock' until a FWP2-orientated shareware disc is produced (if ever). In the meantime, KX-P1081 owners can write to me with a formatted disc and return postage, if they can't wait until then.

Finally, a reminder that I'm at 56 Crescent Drive North, Woodingdean, Brighton BN2 6SN (no phone calls, please), and would be happy to receive hints, problems, wishes and cries for help. **A**

Outline Fonts

- Available now from only £4.70 per font
- Suitable for all applications which can use RISC OS outline fonts
- 'Hinted' so they will look good on ordinary printers, even at small sizes
- Very attractive terms for schools and LEAs
- The Royal and *Tamsin* fonts are shown here

See the **BEST RANGE** at the **BEST PRICES** - ask for our **FREE 30-page Catalogue NOW**

PipeDrivers

From the publishers of *First Fonts* and *Power Fonts*, a new range of useful extensions to PipeDream.

PipeDriver Dot 1 (only £10) is a **must** for every dot-matrix user, giving full control of features such as print size and colour. Other PipeDrivers cater for **Mathematics**, **Foreign Languages** or **Download Typefaces**, and cost £19 (dot-matrix) or £29 (laser).

Please specify subject and printer model.

Please check our prices for all your needs, e.g.

Impression £135* Acorn DTP £125*
PipeDream 3 + free PipeDriver £135*

FREE £10 VOUCHER for Outline Fonts with the above!

Tempest £105* * P&P £3.50 each.

Panasonic KX-P1124 printer £240 + P&P £7

**Ian Copestake
Software**

Dept ARC2, 10 Frost drive, WIRRAL, Merseyside,
England, L61 4XL Tel: 051-648 6287

Prices exclude VAT Official orders and credit cards welcome

Designed, & typeset

Impression is more than a word processor. It can handle all aspects of the final printed result - the text, line graphics, photographs, company logos etc. Yet it can still be used to bash out a single page of text as well as any 'simple' word processor. It is a document processor.

IMPRESSION

entered you do not have to worry about creating new frames or pages.

Graphics frames may contain any sprite (for example images from Scan-Light) or any Draw file. All graphic frames may have the picture scaled within the frame to any degree. In addition the aspect ratio of pictures can be controlled and even locked to any required value.

RISCOS

Impression is one of the first products to take full advantage of the new multi-tasking WIMP based operating system for the Archimedes, so it is simple and intuitive to use - long gone are the days when users had to remember commands, or codes for each operation. Only five main menu options control everything within Impression.

The program is written in ARM assembler so it is very fast and very responsive, and uses the minimum possible RAM space.

Frames

Impression is a frame based page layout system. All objects on the page are held within frames which may be positioned freely anywhere on the page. Frames can contain text or graphics, they may overlap, and may be transparent or have any coloured background. They can have a variety of borders displayed around them and may be arranged in columns to create multi-columned text.

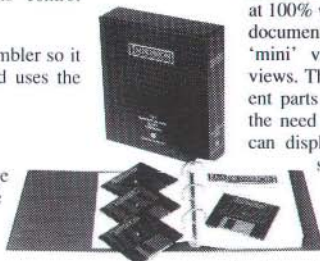
Text frames may be linked to other text frames (even on subsequent pages) so text will automatically flow from frame to frame and page to page. Since Impression has been designed primarily as a word processor, it is important that users can enter text unhindered. Therefore frames and pages are created automatically as text flows out of a frame, so that while text is being

Windows

Impression can handle up to 16 documents in memory at any one time, each being viewed in one or more windows. Each individual view may be scaled as required so that, for example, one view may be at 100% while another window shows the same document scaled to 20% so showing a live 'mini' view or multiple page 'thumb-nail' views. This mechanism also allows two different parts of a document to be edited without the need to scroll between them. Impression can display its pages within the window as side-by-side left/right pages, and as vertically arranged pages in a more word processor-like fashion. There is no need to specifically turn over the page, thereby overcoming a limitation of traditional DTP systems.

Spelling checker

Included with Impression is a 60,000 word spelling checker providing some of the most advanced spelling facilities. Check-as-you-type, user dictionaries, ignore dictionary, crossword and anagram solving and an intelligent 'guess' feature are included. Other related dictionaries control automatic abbreviation expansion as you type, and a hyphenation exception dictionary for precise hyphenation control over and above the normal automatic hyphenation.



arranged on

Styles

Like the most powerful word processors on the Mac, Impression supports a system of styles. Rather than having fixed text effects such as bold, underline etc, Impression allows the user to apply any user named style to any part of the text. This style may be defined and re-defined at any time to represent any combination of stylistic effects. So for example one style, perhaps called 'heading', may specify text to be in a Times font, one inch high, in italics and centered. This style may then be applied to any region of text with one key-press.

Printing

Included are the latest RISCOS printer drivers for Epson compatible 9 and 24 pin printers, and LaserJet compatible laser, ink-jet, and PostScript printers. These printer drivers ensure the output is to the maximum resolution the printer can manage. Impression also supports 'text mode' draft printing so text may be output as fast as possible using the printer's character set. The user therefore has a choice between fast, text only printing or high quality text (any font, any size) and graphics printing.

Impression comes with a 'no quibble' money back guarantee when purchased direct from Computer Concepts.

This advert was designed, entered, laid out and edited on Impression. All logos were created in Draw and imported into Impression frames. The pages were then 'printed' via the PostScript printer drivers to disc. This disc file was then sent directly to a Linotron photo-typesetter, which output the final camera ready artwork. The studio photographs were then pasted over scanned versions.



Other features:

- Master pages for consistent document design.
- All measurements may be specified in inches, mm, cm, points, picas, all to an accuracy of 1/72000ths inch.
- Ideal for a 1Mbyte machine - over 300K free memory!
- Fast automatic disc spooling of documents larger than memory, allowing virtually unlimited document size.
- Print multiple pages per sheet, any scale and rotated.
- Includes Acorn's new outline font system and fonts.
- Fast graphic scale and PostScript graphics printing.
- Automatic index generation.
- The package includes a 200 page manual (produced entirely with Impression), with tutorial, introduction and reference sections. Four discs include Impression, the printer drivers, the outline font manager and fonts, example document, Line-art examples, utility programs etc. An independently produced guide book to typography and page design is also included.

£149.00 +VAT (£171.35 inc)

A more detailed brochure is available from:



Computer Concepts Ltd

Gaddesden Place, Hemel Hempstead, Herts HP2 6EX. (0442) 63933

Introduction to 'C' – Part 6

Chris Dollin

After the manic attack of the last article, we return to a somewhat more sedate pace. This article will tie up some loose ends from our previous articles. We will discuss independent compilation and the use of header files including conditional compilation, concluding with some remarks about program layout.

Carving up the Code

The programs we have discussed so far have been self-contained, apart from using standard C library routines and (last month) the Release-3 C Wimp library. However, in any project involving more than a few hundred lines of code (and in some smaller projects too) some mechanism is needed to split the code up into separate units which can be compiled, and understood, in relative isolation.

C's support for this is less than perfect but it can be made to serve tolerably well. Code can be split into multiple source files, each containing a collection of definitions of types, functions and variables. The source files can be compiled separately into object files which can then be linked into a single program.

It is possible for definitions in one source file to refer to items defined in others, typically by using external declarations that inform the compiler of the existence of these items. While it is possible to scatter these external declarations throughout the source files that need them, good practice dictates that the external declarations corresponding to a particular source file are collected together into a single header file. This header will then be included into source files that require the declarations. Traditionally, C source files have a suffix ".c" (and reside in a "c" directory on the Archimedes) and header files have a suffix ".h" (and reside in an "h" directory on the Archimedes), with the header file and its source file sharing the same prefix (name).

Let's see how this works with a simple example. Suppose that the file "c.main" contains:

```
#include <stdio.h>
#include "service.h"
extern int main( int argc, char
                *argv[] )
```

```
{
    serve( stdout );
    return 0;
}
```

And file "h.service" contains:

```
extern void serve( FILE *output );
```

And file "c.service" contains:

```
#include <stdio.h>
#include "service.h"
void serve( FILE *output )
{
    puts( "You rang, M'Lord?\n",
          output );
}
```

The C files can be compiled separately and linked together and the resulting program will print the message in *serve* when it is run. *main* knows about *serve* because of the *extern* declaration in "service.h"; this declaration looks just like the definition in "service.c" except that there is no function body and the declaration is terminated by a semi-colon.

You might wonder why "c.service" includes "h.service". After all, it is defining *serve* – surely it does not need the declaration as well? The reason we include "service.h" is to allow the compiler to check that the declaration and definition correspond; otherwise it would be possible for all the clients of "service.c" (i.e. other files that refer to objects defined there) to have one idea about the service, and for "service.c" itself to have another, leading to disaster at run-time.

Note that ANSI C permits functions to be called even if they have not been declared, to maintain compatibility with Classic C; the function is presumed to deliver an *int* value, and to take any arguments whatsoever. This can lead to obscure errors: a collection of files that compiles may fail to link when the linker discovers that there is no definition for the function, or may fail to execute because the function has been called improperly. Fortunately the Acorn C compiler will issue a warning when it makes such a presumption. (Experience shows that ANSI C compilers that don't do this are a pain.)

A useful principle to follow is that all header files should be understandable in isolation. By this, I mean that you should not need to know where a header file is included in order to understand its declarations, and a header file should not (normally, at any rate) carry with it provisos of the sort "Files *whatever.h* and *thingy.h* must have been included". Instead, the header should itself include any headers it requires in order for its own declarations to make sense.

In our example above, this means that the includes of *stdio.h* can be removed from "main.c" and "service.c", and placed in "service.h". Note that this does not mean that header files should include all the gubbins needed for their corresponding source files to compile – only those needed for the header file itself to make sense.

"Include Me Out"

Of course, life is not as simple as that. C permits multiple declarations for the same variable or function, so long as they match and the compiler and linker see only one definition for each named item. However, it is not as liberal for type definitions: a second *typedef* for the same type name will be reported as an error.

You might well say "So what? I'm not going to be including files more than once, am I?" Alas, you are. Suppose, for example, "x.h" and "y.h" both include "z.h"; and that "it.c" includes both "x.h" and "y.h". Then "z.h" will be included twice which may cause errors. In a large system, such multiple includes may be essential to keep the program comprehensible and they cannot be omitted. What we need is for the inclusion of any particular file to be idempotent: doing it more than once is the same as doing it just once. All of the ANSI standard header files (with the exception of *assert.h* – see below) are idempotent.

To make idempotent header files, we exploit a facility called conditional compilation. This allows parts of an input file to be thrown away before they are processed by the part of the compiler that deals with type, function and variable declarations. For example:

```
#ifndef FEATURE
extern void use_feature(void);
#endif
```

will be compiled (as a declaration) only if *FEATURE* has been defined by a *#define*. There is also a *#ifndef* (meaning "if not defined") and, as you might hope, there is a *#else*. These constructs can be nested.

Now we can make "service.h" idempotent by wrapping it up in a *#ifndef*:

```
#ifndef H_SERVICE
#define H_SERVICE
extern void serve( FILE *output );
#endif
```

This tests to see if *H_SERVICE* is defined; if it is not, it defines it and supplies all the pertinent declarations. If it is, it throws away the definition and declarations. Of course, for this to work properly all the programmers working on a project must agree to reserve names starting "H_" for these symbols but this is a small price to pay.

Conditional compilation can be a powerful tool for dealing with portability problems. For example, a module may provide a single service that must be implemented in different ways on different operating systems: e.g. reading names from a directory. It can contain multiple definitions for the service with only the appropriate one enabled by a suitable *#ifdef*. To facilitate this, most C compilers provide a way for names to be *#define*-d from the compiler command line: the option *-DName* is used for Acorn (and Unix) C compilers.

Macro parameters

We saw in an earlier article how to use *#define* for defining macros that allowed us to name constants. It is also possible to use it to define macros with parameters; these look like function calls, but do not have their run-time overhead. For example:

```
#define average(x,y) ((x) + (y))
                               /2)
```

There must be no space between the macro name and the open bracket. Leaving a space defines a macro *average* with body

```
(x,y) ((x) + (y)) /2) !!!
```

Of course, the call of *average* may have spaces before the bracket. Wherever *average* appears in subsequent text (but not inside strings) followed by arguments in brackets, it and the arguments, are

replaced by its body, with x replaced by its first argument and y replaced by the second.

You may have wondered at the apparent enthusiasm for brackets we have shown in this example. Because macros are textual replacement and are replaced before the compiler does any expression analysis (which respects C's rules of operator precedence), it is possible for unexpected results to arise. Consider for example:

```
#define twice(x)    x*2
...
... twice( 20 + 3 )
...
```

The call of *twice* will be replaced by $20 + 3*2$, which is probably not what was intended. For macros whose bodies are expressions (which is the greater number), bracketing each appearance of a macro argument where it appears in the body prevents this sad possibility. Similarly, consider:

```
#define plus1( x )    (x) + 1
...
... plus1( 22 ) * 2
...
```

The call of *plus1* and the multiplication by 2 will be turned into $(22) + 1 * 2$, again probably not what was intended. Thus the entire body of an expression macro should be bracketed.

Because the macro processing is independent of the syntax of the rest of the language it is possible to write "syntax macros" which make C look like other languages; for example, defining *IF* to mean *if* (, *THEN* to mean *) {*, and so on. This temptation should be resisted. It makes your C code harder to understand by readers experienced in C and compiler messages will be in terms of the expanded code, not your macros: this can lead to very obscure diagnostics.

The safest use for macros is as "fast functions". Because the appearance of a macro call is the same as that of a function call, it is possible to write code which will use macro expansion or function calls, depending on some compile-time switch (such as the setting of a *#define* and its testing with *#ifdef*). This code can appear in header files, allowing a particular piece of program to be implemented in several different ways without the users of the header file being any the wiser. In general, it is easier to debug code that uses functions. For example, the

Acorn symbolic debugger allows breakpoints to be set on functions, so that execution will be suspended when the function is entered. However, using macros will often make the code faster because the overhead of jumping into and out of the function and having to arrange for the arguments to be available in a standard way can be avoided.

Be warned: using macros in this way can turn round and bite you if the arguments used as expressions have side-effects. Remember that if an argument appears several times in a macro body, it will appear that many times in its expansion and so it can be evaluated many times (unlike a function argument) – embarrassing if it happens to be incrementing a count or outputting a character!

Assertions

A useful programming technique is the use of assertions in your code. An assertion is just a truth-valued expression (the sort you see after *if* and *while*) which the logic of your program requires is *TRUE* (in C, non-zero) at the point where you write it. Assertions provide documentation for the reader and can provide a degree of error protection.

The ANSI C header file *assert.h* provides a macro called *assert* which can be used to include assertions in your code. Normally, the macro generates code which tests the value of the expression at run-time: if it is *FALSE*, the program will stop with a suitable error message (suitable for the programmer, that is – it is not a substitute for proper user-level diagnostics): if it is *TRUE*, the program will continue.

For example, suppose that we were to write a function to find the largest element in an array of integers.

```
#include <limits.h>
int largest( int *numbers, int count )
{
    int largest = INT_MIN;
    while (count > 0)
    {
        if (*numbers > largest) largest
                                = *numbers;

        count -= 1;
        numbers += 1;
    }
    return largest;
}
```

Now, it is not clear what this function means when `count <= 0`; it will deliver `MIN_INT` (which is defined in `limits.h` to be the minimum value for an integer in this system), but this is an artefact of the implementation. If we document the requirement that `count > 0`, we can check it with an assertion placed just before the `while` loop:

```
assert ( count > 0 );
```

A correct program will not violate any of its assertions. If, in the course of development, it turns out that the cost of checking the assertions is too high, defining the symbol `NDEBUG` before including `assert.h` will stop `assert` doing any run-time checks.

Unlike all the other ANSI headers, `assert.h` can be included multiple times and will produce different definitions for `assert` depending on the definedness of `NDEBUG` at the time it is included.

Laying out the law

I have not said anything about program layout so far in these articles, preferring instead to demonstrate my own style in the examples. However, it is not the

only nor indeed the most common, style and since verbal wars have been fought about program layout, a few words are in order here.

There are three rules to remember. First, pick a style and stick to it. It will be easier for you (and other readers) to understand your code if you have a consistent style. (But don't be afraid to experiment for a while until you find one you like.) Second, do not be afraid to use spaces to aid legibility. Third, use indentation to show structure – it helps to see what is controlled by what.

After that, you're on your own. If you look back at our examples you can probably guess the rules I use for my own layout.

exit

I hope to base future articles on problems – or techniques – submitted by readers. Next month I will discuss the Mystery of the Missing `bbc_Rnd` (thanks to Mark Taylor). If you have any questions, problems, comments or suggestions, please send them to Paul or me c/o Archive. **A**

PipeLine

Gerald Fitton

Get Your Free PipeDream3!

Well, not exactly PipeDream3, but Colton Software's PipeDream3 Demonstration Disc. I haven't yet discovered all the things that you can't do with the demonstration disc but you certainly can't save your work. It does give a very good indication of how PipeDream works. I have included it, with Colton's permission of course, on the monthly disc.

For those of you not wishing to buy the Archive monthly disc, I have a few copies of the demonstration disc that I can lend you. The arrangement is this: you send me a jiffy bag and a label with a stamp. I send you the demonstration disc and you return it to me when you've tried it out so that I can let someone else have it.

When you have decided that you can't live without PipeDream, you order the full version from Archive at £145 (or direct from Colton at £147 +VAT).

Anagrams and more bubbly

Printing anagrams and subgrams isn't possible. Printing a dictionary is. Dump the dictionary to a

file using `^#` as the template. Then print the file! Don't infringe Colton's copyright or we'll all be in trouble.

The competition is to find the word which has the most anagrams included in Colton's PD3 dictionary. To get you started you might try 'teas'. Next month I'll give you a word which has eight anagrams (including the original word) – no! I didn't think of it myself – so you'll have to beat eight. However, any reader who sends me a word of seven or over will get an honourable mention in the next edition of PipeLine.

To win the bottle of bubbly you must send your word (more than eight anagrams including the original word) direct to Robert Macmillan at Coltonsoft Ltd quoting your PD3 registration number (found from clicking <menu> then info on the installed PD icon).

Subgrams are also an interesting pastime. Find a long word in Colton's dictionary and see how many subgrams you can get. There is no prize for this but if you send some good ones to me I will publish the best in next month's Archive.

Bubbly for formatting to two widths

Up to now Colton report "We have not received many responses and those that have arrived either solved a different problem or cheated. By cheating, I mean so that it (the text) becomes completely unformatable." I've solved it, so it can be done.

Hard space

Talking of space, character 160 is a hard space of fixed width. Insert it by holding down the <Alt> key and typing '160' on the numeric pad. When you release the <Alt> key, the space appears. I have had a few readers who like to see their hard space. If you do, then type in some little used character as your hard space and then either search and replace with character 160 (from the spell menu) or translate by modifying your printer driver translation list.

Non PipeDream users and 1M Archimedes

Whether or not you are a PD user, send me an s.a.e. that will take an A4 sheet and I'll send you a copy of an application note from Colton explaining what they think is the best way of handling the problem of memory shortage on 1M machines.

You may wonder why non-PD users will be interested. The answer is that the problem of memory shortage is not exclusive to PipeDream, nor (if it comes to that) is it caused by PipeDream. If you want a sophisticated piece of software that gives you many of the features of a desk top publisher with the ease of use of a wordprocessor, plus a first class spread sheet and a reasonably good database thrown in for good measure (i.e. PipeDream 3) plus fonts and RISC-OS printer drivers and maybe a graphics package (!Draw or !Graphbox) then you really need 2M of memory. Upgrades to the 310 are available now at a price, and that is the route I strongly suggest you take.

Row numbers

Insert a new column to the left of the 'A' column so that it becomes the new 'A' column. Now press Edit Expression, <F2> and type in "row" and press <return>. Replicate this expression down the 'A' column as far as you want to go. Hey Presto!: row numbers.

Greek and other outline fonts

I have been told that Beebug (who?) have some Greek outline fonts. Also, Ian Copestake has sent

me a sales leaflet about fonts he has available but I am not sure from the documentation whether these are outline fonts or whether they are printer specific and redefine what is printed at the printer when the printer receives character codes from, say, a PipeDream printer driver.

Let me try to clear up one misunderstanding. PD3 will print out via the RISC-OS printer drivers as well as via the PipeDream printer drivers. The RISC-OS printer drivers are those which, up to now, have been provided by Acorn and you use the RISC-OS drivers with anti-aliased or their recent replacement, outline fonts. I strongly recommend anyone not using the PipeDream printer drivers (i.e. using the RISC-OS drivers) to change from anti-aliased to outline fonts if they have not yet done so. A starter set of outline fonts is available through Archive at £47 and other outline fonts are becoming available at about £5.00 to £10.00 per font from, amongst other places, The Electronic Font Foundry, 18 Brockenhurst Road, Ascot SL5 9DL. (*Beware that you need the !Fontmanager which is NOT PD and is only available, as far as I know, with the Acorn Font Packs, Acorn DTP and, under licence, various other DTP's. Ed.*)

Delete to end of slot

If you have a multi-column format and you want to get rid of a line in one column then 'Delete Row in Column' works but it pulls everything in that column up a row and this might mess up your table. You should use 'Delete to End of Slot' instead. If you want to delete many lines in a column but don't want the column to move up then don't use 'Block Delete'. Use 'Delete to End of Slot' in the first row and then 'Block Replicate' (<Alt>+BR) to replicate the blank slot down through the column.

Labels (again)

Again, to clear up a misunderstanding, make sure that you set the page length to the vertical distance between labels.

Insert on return

If you are working on a spreadsheet make sure you turn this option 'Off' otherwise you will get in a mess that will take hours to put right. Remember to turn wrap (and justify) off when working with a spreadsheet.

To CR + LF or not?

This refers to the use of PipeDream printer drivers. Make sure that you use LF (line feed) and CR (carriage return) in such a way that everything works properly. This problem can take ages (or pages) to explain but essentially you need to have your *Configure Ignore option, your printer's dip switches and your printer driver option set to a compatible set of commands.

My own settings are: *Configure Ignore (i.e. ignore nothing) which gives for *Status - No Ignore, the printer dip switches so that there is no automatic line feed (i.e. the computer must send the LF character) and my printer driver option so that it does send the LF character to the printer. There are other combinations that work equally well. If there is some clever reader out there who has the whole set of 'workable' options on a disc that they are willing to send me than I shall be pleased to make them available to anyone with problems.

Unwanted form feeds

This is a similar problem to the one above, i.e. of getting the hardware and software configured to a compatible set. I am still struggling to get my laser printer to produce labels from a 'mail merge'. I get a form feed after each label and I can't seem to get rid of it! I have solved the problem with my FX80 using a PipeDream printer driver by taking out the form feed from the printer driver but I am still having problems with the RISC-OS driver because I can't find out how to get rid of the form feed. Help!

Loss of right justified text

If I had a £1 for every letter I've had from people with this problem then I could afford to buy the stamps to reply to you all and still have a week's holiday! Please accept my apologies for not replying to each of you individually. If you are losing right justified text when you print then it is almost certainly because you have a right margin position (the little vertical arrow set with <Alt>+H) which is either (a) off the right hand edge of the paper you are printing on or perhaps (b) outside your range of columns (i.e. beyond the rightmost column). The solution is (a) to use <Alt>+H to reduce the right margin position or (b) add another column. It is important to follow this by reformatting your text before trying again.

RISC-OS printer drivers

Somewhere, I have seen a RISC-OS printer driver advertised for the Canon BJ-130, 360 dpi printer at about £10.00. It might be 'Font Foundry' but I'm not sure. If anyone can help, please let me know. I'm still looking for an Epson GQ-3500 RISC-OS driver so that I don't have to work in HP Emulator mode all the time. The HP DeskJet seems to be a popular RISC-OS printer from the amount of correspondence I get printed out using one. I cannot distinguish between the output of the DeskJet and the LaserJet. Would any reader care to comment?

The monthly program disc

In addition to the 'free' PD3, there are three more sub-directories of PipeLine on the Archive monthly program disc.

Jim Ducat has sent in PipeDream files for a Cassette Inlay Printer and "a number of spreadsheets for customising, including a financial planner, a building society account manager and a car costs monitor". The building society spreadsheet shows how to insert days of the week automatically into a spreadsheet.

John Greening has sent in some utilities, with 'readme' documentation, for converting Mini-Office II files to PipeDream. This is a once-only operation and very worthwhile if you have files in Mini-Office II format.

Barry Allen has sent in PipeDream printer drivers for the HP DeskJet and Panasonic KP 1081 which will print out the IBM graphics set together with a PipeDream key file and a modified PipeDream !Run file which lets you enter IBM graphics so that they appear on the screen.

Background printing

Here's a useful tip I have been sent by Colton: "Having a fast computer and a slow printer leads people to get very frustrated when the computer gets tied up talking to the printer and the user doesn't get a look-in until it's finished. People ask for background printing. Well, we already have it in the RISC-OS drivers !PrinterDM and !PrinterPS and, hopefully, in the RISC-OS drivers being written by other people. We first get PipeDream to do its print operation but, rather than sending the output straight to the printer, we redirect it to a disc

datafile (because this is faster than sending the file to the slow printer). Then we redirect the driver to send to the printer and drag the datafile from the disc to the printer driver icon. This operation will work in background, enabling us to carry on working in the foreground. So we have made the (slow) printer-dependent operation into a background task. Note that the first stage of the operation (output to a disc datafile) can be done with either the RISC-OS driver or the PipeDream driver."

One word of warning: RISC-OS drivers generate enormous files. A typical page for me generates a 1M size file so I can't get it onto an 800Kb floppy. However, for half pages, I can manage to get the 500Kb file onto an otherwise blank floppy!

I have a 4M machine so I created a RAM disc of about 1.5M and redirected my RISC-OS output to that. It seemed to me to work better than when I used the hard disc.

Thanks to you!

Keep those letters coming in. I hope that, as the months (and years?) roll by to get more in tune with what you want in this column. At present, it seems you prefer a mixture of hints and tips such as we have here, plus a very simple article such as the multiplication table of February, leaving the more complicated things (e.g. PipeDream printer drivers, spreadsheets and data base examples) on disc with adequate documentation. I hope that as we all get to know PipeDream better we shall get some more extensive examples needing linked or cross-referenced files dependent documents and macros. A few comments from the more expert reader (you?) on how you think PipeDream's linked or cross-referenced files with dependent documents compares with a full relational database would be a welcome contribution. Have you an example of a problem solved by such techniques? **A**

Choosing a Laser – The HP Option

Robert Hamilton

As the price of laser printers falls, a good number of Archimedes and A3000 owners will be taking advantage of the quality and speed which these printers offer. There are two laser printer drivers available for RISC-OS at present, one for PostScript lasers and one for Hewlett Packard LaserJets. (*Available on the RISC-OS Extras disc – Shareware 17.*)

As most PostScript printers cost from around £3000 upwards, this review concentrates on HP LaserJet compatibles, which generally cost between £900 and £1600. There are speed differences between the two types but for most purposes, the output looks exactly the same. Running costs are also similar, usually around two pence per sheet.

If your budget means that you can consider a PostScript printer, don't be fooled into thinking that buying one will automatically give you the best possible machine for your needs. Before you choose between the two standards, you should think carefully about what you will be using the printer for. This is because some features are faster on PostScript and some are faster on LaserJets.*

Text printing

If your output is mainly textual, the choice should

be made on the basis of which set of fonts you will be using most – those built into the printer or the RISC-OS anti-aliased fonts.

If you use the RISC-OS fonts, a PostScript machine will be a lot quicker than a LaserJet. This is because PostScript printers have a built in range of scaleable fonts, and RISC-OS usually only needs to provide the printer with details of a character's typeface along with its size and position on the page.

For high quality output on a LaserJet however, RISC-OS provides all fancy fonts as bit-mapped graphics. This takes both time and memory, as producing a 300 dots per inch bitmap definition of a character is a complex task! Having to send all this information to the printer also slows things down. Some software lets you choose which set of fonts (RISC-OS or printer) to use (most notably PipeDream 3), but more often than not you have no choice but to go along with whichever approach the package supports.

If you are only sending pure text to the printer (i.e. not using the anti-aliased fonts) such as output from First Word Plus or program listings, then you will hardly notice the difference between PostScript printers or LaserJet machines. Without indulging in

PostScript programming, you will not be able to use the extra fonts easily (thirty-five are supplied as standard with the language), and the output from both printer types will be good.

Some of the LaserJets have more fonts built in than others and you can make use of these quite easily from word processors and similar programs. However, they are usually only supplied at certain point sizes, and using the printer to scale them to other dimensions can produce poor results. This means that if you want to use very large headings or similar effects without resorting to using the RISC-OS fonts, you may need to buy additional font cartridges to plug into your printer.

Printing text and graphics

If your output is more complex, then so is choosing a laser printer! On the whole, LaserJets output sprites faster than PostScript and the increase in speed can be a factor of five or more. The fact that the former are slower for fancy font output makes it difficult to find a compromise. Computer Concepts' Impression document processor produces hard copies more quickly than most programs when sprites need to be printed, and the speed differences are less noticeable with this package.

Lineart is usually quicker under PostScript because the Draw module itself is written to work well with Adobe's page description language. The printer itself will work out the Bezier curve routines and create a bitmap version for use on paper. With LaserJets, the host computer does all this work instead: the calculations themselves are probably done more quickly by an Acorn RISC than by a laser printer, but the fact that more data needs to be transmitted means that the overall printing process is held up. If you can, try out a page of your 'typical' output on both types. This will demonstrate the speed differences between them and may help you to decide.

This review was based on using the printers with a variety of RISC-OS software, Acorn Desktop Publisher, PipeDream 3, Paint and Draw. These are the main programs which output to printers, and I also used First Word Plus for testing.

When printing from First Word Plus, the fonts within the printer are used, whereas the other programs rely on fonts provided by the RISC-OS

machine. This can lead to problems as a full page of graphics takes a megabyte of printer memory. So, if you want to produce a full page of DTP, sprites or even smart output from PipeDream 3, you'll need at least a megabyte of printer memory.

When the printer runs out of memory, it prints what it can (around 2/3 of an A4 page if you've only half a megabyte), does a form feed, and prints the rest on another sheet. This is the best of a bad job but unless you are going to use your laser exclusively for word processed output from programs like First Word Plus, you will need at least a megabyte of memory. For most of the lasers, an upgrade costs around £500 for an extra megabyte, so you actually end up with a 1.5M printer.

The Hewlett Packard LaserJet

This is the printer which all the others emulate, so it should be good! It is a very well-built machine and has the usual paper tray poking out from its front panel. It arrives set up but getting it going should not be a problem as the procedure used is very similar to that of the Star. I experienced very few problems with the machine and it was extremely reliable. The one annoying occurrence was paper jamming – one slight jam often led to a series of others. However, this did not really detract from the machine and if you want the reassurance of the standard, the HP LaserJet is a good machine. As you would expect, Acorn's LaserJet driver controlled the printer very well and people to whom I showed it said that the output was as good as that from a PostScript machine. Although this is as much a reflection of the Acorn printer driver as this particular printer, it shows that where Archimedes systems are concerned, HP LaserJets are not always the poor relations of PostScript printers!

The Hewlett Packard DeskJet Plus

This printer is not a true laser, but it belongs here because it is LaserJet compatible and works with the RISC-OS drivers. Its inkjet printing system is much slower than the other machines, being rated at around 4 pages per minute as opposed to 6-8 ppm for the lasers. However, it is very difficult to compare speeds, as four pages per minute can really only be achieved for pure text output with a generous following wind: in truth the DeskJet Plus is a bit quicker than the fastest twenty-four pin printers.

As you probably know, a laser printer only achieves the stated ppm speed when it is doing multiple copies of the same page or when it is only producing text output. Five different pages can take much longer to print than five copies of one page, because each different page has to be built up in memory. The DeskJet Plus does not use memory mapped graphics so it takes almost the same time in both cases. This means that if you want to print lots of copies, it's probably easier to print once and then photocopy the rest of the run. On the quality front, the DeskJet Plus does quite well. In a quick poll around the office, people did not rate its output as highly as that from an Apple LaserWriter or that from an HP LaserJet II, but they did say that it was very good and certainly acceptable for business correspondence.

A very useful feature for use with character based programs was a button that allows you to output in landscape format, i.e. with the A4 page sideways. The rotation is handled by the printer and I used this option a lot with Logistix, which cannot normally rotate output to LaserJet printers. It is also useful for outputting program listings which include very long lines.

The DeskJet Plus produces near laser-quality output which is only just distinguishable from that of the true lasers and yes, there is a catch! The ink it uses is water soluble, so it tends to smudge if you rub it or if it gets a little damp. This isn't too noticeable in dry conditions but you can easily tell if a letter was printed on a DeskJet by rubbing a moistened finger over it. As the printer uses a liquid ink system, printing large dark areas often leads to the paper becoming wet and wrinkled. Also, the pages come out face up (via a very clever system of levers and catches) so you may need to sort them after printing. If you can live with these limitations or if you are thinking of buying a 24 pin printer, then the DeskJet Plus is for you. Do remember that it's only an option if you don't need the full speed of a laser and it is probably fairer to compare the DeskJet with top-end 24-pin machines.

Note: The HP DeskJet is the forerunner to the DeskJet Plus and it's still available, often for a very good price. It's quite a lot slower – text is printed only half the speed of the Plus and graphics take five times as long as with the upgraded design.

The Qume CrystalPrint Series II

The CrystalPrint range is the upgradable option – you can start with a basic machine and add to it until you have a PostScript compatible printer. However, using it with a RISC-OS machine is problematic. One morning I unplugged a working LaserJet II, plugged in the Qume instead and spent the next half hour working out why it wouldn't work. In the end, I found that its margins were different from those on the HP LaserJet so it was not 100% compatible in use. It also had a strange habit of printing a blank page after each print run. Again, this was down to a control code which did nothing on Hewlett Packard machines but caused problems with the Qume.

These problems were very erratic and other people who used the printer often found strange 'features' which could never be reproduced! On the plus side, the Qume has a very small footprint as its paper tray fits almost right inside the printer. Should you buy one? If you know that you will want a PostScript printer but can't afford it yet then it seems the Qume is your only option. However, if you're looking for a straight LaserJet, then there are better machines on offer.

The Star Laser 8

This is the only printer which comes with a megabyte of memory as standard and this reflects the overall quality of the machine. It seems to have the same engine (a Canon) and body as the HP LaserJet II, but Star have added some extras. The control panel on the printer allows an amazing array of features to be controlled – even the brightness of the LCD display itself can be adjusted! Using the Laser 8 with programs like First Word Plus is very easy as it has a nice range of built-in fonts and any of these can be selected as the default.

This was also possible on the LaserJet II but the Star's front panel is just one of the bonuses which makes it such a good printer. I experienced no compatibility problems with the machine and it produced excellent output from the RISC-OS drivers. Black areas were very solid and dithered output excellent. As with the HP LaserJet II, its output was rated equal to an Apple LaserWriter's efforts by almost everyone who examined it. During a month's heavy use, the Star never suffered a paper jam. In fact, I had no problems with it at all.

This reliability and the fact that 1M of memory is the standard makes the Star the pick of the bunch as far as LaserJet compatibles are concerned.

Other machines

There are now a few low cost laser printers which provide a throughput of around four pages per minute. If these are HP LaserJet compatible, then they will always be worth a look especially as the price divide between them and machines like the DeskJet is narrowing all the time. There are some very nice machines to be had within this price range and shopping around for bargains can provide you

with a way of producing very high quality hard copies.

**The other option is to have a completely dumb (and therefore cheap) laser engine and put the intelligence in a podule on the Archimedes. Using this approach, Computer Concepts have produced what promises to be a very cheap laser printer for the Archimedes which operates at extremely high speeds, even compared to the Apple Laserwriter NT2 which CC used to use for their DTP work before they developed their own hardware and software. Ed. A*

Calculating Prime Numbers

Paul Denize (New Zealand)

The article by Brian Cowan (Archive 3.4 p29) "Mathematical Applications" brought to our notice the difficulty of finding the Nth prime number. The problem was not greatly difficult until pondering the possibility of the millionth prime.

The memory required to store all the 1 million, 32 bit, primes would require 4M. Then there is the problem of speed – Brian estimated 3 days. The solution I came up with solves both these problems up to the 1 Millionth Prime. The program took 2 hours 45 minutes and used about 500k of data.

Primes of, say, order 10 higher begin to deal with greater than 32 bit arithmetic and devising some form of "memory paging" or "stepwise solution".

Memory

The 1 Millionth Prime is 15485863 as stated by Mathematica. If a bit array 16 million long is used to represent the numbers that bound the first million primes then all we have to do is clear all the bits that aren't primes. Thus requiring only 2M of memory. We all know that multiples of two aren't prime so if only the odd numbers were stored then only 1M is required. I played with a few calculations and determined that the best saving is produced by not storing multiples of 2, 3, 5, 7 and 11 any more cost too greatly in the table to determine the bit address and fewer lost the memory battle.

The way I implemented the bit array is given below:

```
REM NOTE : 2*3*4*5*7*11 = 2310
DIM Known_Prime%(5)
```

```
Known_Prime%(1)=2
Known_Prime%(2)=3
Known_Prime%(3)=5
Known_Prime%(4)=7
Known_Prime%(5)=11
DIM Table_to%(2310-1)
FOR I%=0 TO 2310-1
    Table_to%(I%)=0
NEXT I%
FOR I%=1 TO 5
    FOR J%=Known_Prime%(I%) TO 2310
        STEP Known_Prime%(I%)
            Table_to%(J%-1)=-1
    NEXT J%
NEXT I%
J%=0
FOR I%=0 TO 2310-1
    IF Table_to%(I%)=0 THEN
        Table_To%(I%)=J%
        J%+=1
    ENDIF
NEXT I%
```

The Table_to% array now contains a number from -1 to 479. The table provides an Index from a 2310 bit word shrunk to one of only 480 bits. To get from real bit N, the Table_to array is used if the index supplied is -1 then there is no mapping as the number is already determined as not being a prime. This gives us the size of the bit array required. If Prime(1000000) is to be reached then 16000000 bits must be mapped requiring:

```
Bytes% = ((16000000 DIV 2310) *
          480) DIV 8
```

About 500k !

Calculating Prime Numbers

A special note must be made that the bits for 2, 3, 5, 7 and 11 are not mapped and will not be primes so we must take care of these separately. Also, bit(1) will not have been masked out.

Speed

The next step is to speed the process up by minimizing the calculations. The simplest form would be:

```
FOR I%=2 TO SQR(16000000)
  FOR J%=I%*I% TO 16000000 STEP
    I%
    clear_bit(J%)
  NEXT I%
NEXT I%
```

Two things can be noted: firstly, we need go no further than the square root of 16000000 and secondly that $I\%*I\%$ is the lowest uncleared non-prime.

This, however, is wasteful – many bits are cleared twice. The solution is only to remove the bits that are multiples of numbers that are not multiples of others (i.e. still considered primes) and between the two set bounds.

```
I% = 2
WHILE I% < SQR(16000000)
  J% = previous_prime_to
    (16000000 DIV I%)
  WHILE J% >= I%
    clear_bit(I%*J%)
    J% = previous_prime_to (J)
  ENDWHILE
  I% = next_prime (I%)
ENDWHILE
```

Note that $I\%$ steps only from prime to prime and that $J\%$ steps in reverse. This ensures that $I\%$ to higher powers are removed also. Each bit is removed only once.

On completion, one needs only find the 1 millionth bit set and calculate back the real bit number. 2 hours 45 minutes later... 15485863!

```
10 REM > Primes
20
30 INPUT "NUMBER ? " Request%
40
50 TIME=0
60 DIM KnownPrime%(5)
```

```
70 KnownPrime%(1)=2
80 KnownPrime%(2)=3
90 KnownPrime%(3)=5
100 KnownPrime%(4)=7
110 KnownPrime%(5)=11
120
130 IF Request%<6 THEN
140   IF Request%<1 THEN
150     PRINT "Prime(";Request%;") is
                                   Undefined."
160   ELSE
170     PRINT "Prime(";Request%;") is
                                   ";KnownPrime%(Request%)
180   ENDIF
190 ELSE
200   Primes% = Request%*16 : REM
    approx 1/15+ numbers are prime
210   PRINT "Initialize ..."
220   PROCinitializebits
230
240   REM cant clear primes 2,3,5,7
    or 11 bits dont exist
250
260   PRINT "Calculate ..."
270   I%=13
280   WHILE I%<SQR(Primes%)
290     PRINT I%
300     J%=FNprevprimeplus(Primes%
                                   DIV I%)
310   WHILE J%>=I%
320     PROCclrbit(J%*I%)
330     J%=FNprevprime(J%)
340   ENDWHILE
350   I% = FNnextprime(I%)
360 ENDWHILE
380 PROCclrbit(1) : REM clear 1 last
    as it protects PROCprevprime
390
400   PRINT "Show Result ..."
410   P%=5
420   Words% = ((Primes% DIV 2310) *
                                   480) DIV 8
430
440   W%=0
450   REPEAT
460     B%=0
470     REPEAT
480       IF (Space%?W% AND (1<<B%))
                                   <>0 THEN
490       P%+=1
500       IF P%=Request% THEN
```

```

510      I%=(W%*8+B%) DIV 480)
          *2310 + Table_back%((
          W%*8+B%) MOD 480) +1
520      PRINT "Prime(";P%;" ) is
          ";I%
530      ENDIF
540      ENDIF
550      B%+=1
560      UNTIL (B%>7) OR (P%=Request%)
570      W%+=1
580      UNTIL (W%>Words%) OR (P%=
          Request%)
590 ENDIF
600
610 PRINT TIME;" CentiSeconds"
620 END
630
640 DEF PROCinitializebits
650 LOCAL I%,J%,Bytes%
660
670 REM Lookup Table to the Prime Bit
    Set (-1) = Not Prime & Not Stored
680 REM Lookup Table to get back to
    real bits from the Prime Table
690
700 DIM Table_to%(2310-1)
710 DIM Table_back%(480)
720
730 FOR I%=0 TO 2310-1
740     Table_to%(I%)=0
750 NEXT I%
760 FOR I%=1 TO 5
770     FOR J%=KnownPrime%(I%) TO 2310
          STEP KnownPrime%(I%)
780         Table_to%(J%-1)=-1
790     NEXT J%
800 NEXT I%
810
820 J%=0
830 FOR I%=0 TO 2310-1
840     IF Table_to%(I%)=0 THEN
850         Table_to%(I%)=J%
860         Table_back%(J%)=I%
870         J%+=1
880     ENDIF
890 NEXT I%
900
910 REM - The Prime Bit Set
920
930 Primes% +=2310-(Primes% MOD 2310)
940 Bytes% = ((Primes% DIV 2310) *
          480) DIV 8
950 DIM Space% Bytes%
960 FOR I%=0 TO Bytes% STEP 4
970     Space%!I%=&FFFFFFFF
980 NEXT I%
990 ENDPROC
1000
1010 DEF PROCclrbit(N%)
1020 LOCAL Bit%,W%,B%,I%
1030 I% = Table_to%((N%-1) MOD 2310)
1040 IF I%>=0 THEN
1050     Bit% = ((N%-1) DIV 2310)*480
          + I%
1060     W% = Bit% DIV 8
1070     B% = Bit% MOD 8
1080     Space%?W% = Space%?W% AND NOT
          (1<<B%)
1090 ENDIF
1100 ENDPROC
1110
1120 DEF FNreadbit(N%)
1130 LOCAL Bit%,W%,B%,I%
1140 I% = Table_to%((N%-1) MOD 2310)
1150 IF I%=-1 THEN =FALSE
1160 Bit% = ((N%-1) DIV 2310)*480 +I%
1170 W% = Bit% DIV 8
1180 B% = Bit% MOD 8
1190 IF (Space%?W% AND (1<<B%))=0
          THEN =FALSE ELSE =TRUE
1200
1210 DEF FNnextprime(N%)
1220 LOCAL Bit%,W%,B%
1230 Bit% = ((N%-1) DIV 2310)*480 +
          Table_to%((N%-1) MOD 2310)
1240 REPEAT
1250     Bit%+=1
1260     W% = Bit% DIV 8
1270     B% = Bit% MOD 8
1280 UNTIL (Space%?W% AND (1<<B%))<>0
1290 =(Bit% DIV 480)*2310 + Table_
          back%(Bit% MOD 480) + 1
1300
1310 DEF FNprevprime(N%)
1320 LOCAL Bit%,W%,B%
1330 Bit% = ((N%-1) DIV 2310)*480 +
          Table_to%((N%-1) MOD 2310)
1340 REPEAT
1350     Bit%-=1
1360     W% = Bit% DIV 8
1370     B% = Bit% MOD 8
1380 UNTIL (Space%?W% AND (1<<B%))<>0
1390 =(Bit% DIV 480)*2310 + Table_
          back%(Bit% MOD 480) + 1

```

```

1400
1410 DEF FNprevprimeplus(N%)
1420 LOCAL Bit%,W%,B%
1430 WHILE Table_to%(N%-1) MOD
                                2310)<0
1440     N% -= 1
1450 ENDWHILE
1460 Bit% = ((N%-1) DIV 2310)*480 +
          Table_to%(N%-1) MOD 2310)
1470 W% = Bit% DIV 8
1480 B% = Bit% MOD 8
1490 WHILE (Space%?W% AND (1<<B%))=0
1500     Bit%-=1
1510     W% = Bit% DIV 8
1520     B% = Bit% MOD 8
1530 ENDWHILE
1540 =(Bit% DIV 480)*2310 + Table_
          back%(Bit% MOD 480) + 1

```

Brian Cowan adds....

My article on primes certainly stirred up a lot of interest. As pointed out by many correspondents, what I described was NOT the sieve of Eratosthenes. The sieve method, as described in the article above, consists of simply striking out multiples of each prime in turn. It involves no division and therefore is much faster as well as using much less memory than my method.

A number of people have sent me interesting prime number programs and I hope to include some of these in a future article. In the mean time anyone interested in speed should have a look at the assembler programs in Peter Cockerell's book on the ARM. (*Or Bjørn Fløtten's article on last month, page 51!! Ed.*) **A**

Language Forum

David Wild

The best language?

From time to time the computer press conducts discussions about the languages we ought to be using to write our programs, and this month it seems to be the turn of A & B Computing. In it there is the argument that C is the only language to use, because this is the language in which Unix is written.

When you analyse it, this is an extremely weak argument as any computer only understands its own machine code and it is the job of a compiler to turn the instructions which you write in C, Pascal, Fortran or BASIC into machine code. Once that conversion has taken place, the original language has no relevance at all. The fact that Unix was written in C does prove that it is possible to write substantial programs in C, just as Donald Knuth's "Tex" program proves that it is possible to write substantial programs in Pascal; but there has never been any real doubt that this was the case.

I am sure that, with the exception of a small number of very specialised programs, almost any program could be written equally well in any of the major languages without any user being aware of the difference. The group who really do need to worry are those charged with maintaining the programs and I feel that C tends to tempt programmers to write

impenetrable code while Pascal, being more wordy, is usually much easier to follow.

While it doesn't make much sense to argue about which is the "best" language it can be worth arguing about the quality of compilers and what can be done to improve them.

The best compiler?

A good compiler generates compact, fast code and generates it quickly. This seems to be a reasonable definition but, when we examine what is going on, it turns out that things are not quite so simple. If you study the article on efficient machine code in the March "Archive" you will see that the fastest code is that in which there are fewest branches. This means that there will be more lines of code because repeated instructions may be quicker than loops. Similarly, a fast compiler may not generate optimum code because more passes over the source may make for more efficiency.

It is interesting to note that when I was programming mainframe computers some years ago, many of the manufacturers offered more than one compiler for the same language, depending on the characteristics of the generated program. An article in the March issue of "Byte" suggests that RISC machines need superior compilers to take advantage of the "pipelining" facility.

A good compiler should compile all the language that is in the appropriate standard, and should allow the flagging of any code that is not in accordance with the standard. Turbo Pascal, on the PC emulator, lets itself down badly by not letting the compiler flag non-standard code, while Cambridge Pascal misses out one or two of the requirements of the standard, although it does allow the extensions to be flagged. ISO-Pascal and release 3 of C both seem to conform closely to the standard (or draft) and flag any extensions.

Any compiler should be bug-free but all large programs tend to contain bugs and our best hope is that they will be consistent so that they can be detected and avoided. Minor bugs, such as the one I reported in my review of Cambridge Pascal, where the compilation did not continue after certain errors, don't really matter at all as they don't affect the generated code.

Documentation ought to be good, but many compilers fall down badly in this respect. Acorn Pascal is very bad, with far too few worked examples and a poor layout. I recently had a letter from someone who was trying to run a program by typing:-

```
*prog hellow
```

instead of

```
*hellow
```

and, when I looked at the page of the manual, I could see exactly why he had fallen into the trap. Cambridge Pascal has quite a lot of omissions from the manual and even release 3 of C still has gaps, at least to someone who is unfamiliar with the language.

It would be helpful if compilers could be multi-tasking, especially if compilation will take some time, so that you can be doing something else while they are getting on with the job. Quite often the problem is the length of time the machine is unavailable rather than the total compilation time. So far, none of the Archimedes compilers provides this.

Extensions to the standard can be useful but some, on all compilers, seem to be there because somebody saw a way of providing them rather than meeting a need. Cambridge Pascal provides conditional compilation and, although the example program is rather contrived, it could be useful for

generating programs with different array sizes for different machines. Diagnostics should be good, and it might be a good thing if a compiler, on detecting an error, were to ask the user whether it should insert such things as missing semicolons and continue the compilation.

I would be interested to hear your views on what makes a good compiler and the improvements you would like to see. Perhaps we don't pay enough for our compilers and, because of this, don't demand enough.

Release 3 bugs

Nigel Walker has sent me details of three small bugs in release 3 of C.

1. >bbc-inkey() won't work with negative arguments.
2. >bbc-vpos() always returns zero.
3. If you attempt to generate debugging information, using option -g, for a file which contains recursively linked data structures like:-

```
typedef struct a;
typedef struct { a *link } b;
typedef struct { b *link } a;
```

the compiler crashes.

Cambridge Pascal

In my review, I criticised the fact that some of the demonstration programs could only be stopped by rebooting the machine because they contained loops with 'repeat until false'. The manual explains that there is quite a lot of extra work needed to detect the escape key at any time during the program's run and so it is not detected unless you specify it.

If you define:-

```
function button : boolean;
var
  r1,r2 : integer;
begin
  swi(6,122 to r1,r2);
  button := r2 = 105
end;
```

you will find that 'repeat until button' will allow you to detect the pressing of the escape key at the end of each pass through the loop and this is what is required in most cases. **A**

Arithmetic with Long Integers

Fred Hartley

All of us have indulged at some time in number puzzles. For some, numbers and their properties are a serious object of study. Number Theory is quite abstruse and very strange to those not steeped in it. In many areas of interest no-one has yet produced a theory and such questions are often a happy hunting ground for empirical analysis by amateurs. Sometimes relationships can be established empirically which point to the shape of a theory. Advance by guided trial and error is an accepted method and graced with the respectable name of heuristics.

Numerical problems often require reams of calculations and the time taken to produce an answer depends, as always, on the efficiency of the algorithm and particularly the speed of the machine. Input/output and disc access speeds rarely cause any problem. The arrival of the Archimedes put a fast computer within the grasp of the enthusiastic amateur mathematician.

Integer arithmetic can be a problem

For one large class of problems only integer arithmetic is required since exact answers are essential. There is, however, a 'fly in the ointment'. Frequently, the sizes of the numbers far exceed the capacity of a 32-bit word and multiple length arithmetic has to be used.

It is possible to perform multiple length integer arithmetic, albeit painfully slowly, in BASIC. One such method stores the numbers as decimal strings and operates on them one decimal digit at a time. A two's-complement binary number can be spread over several 32-bit words and calculations carried out one byte at a time. Attempts to use a 'digit' exceeding one 8-bit byte become very messy because it is not possible, for example, to obtain the product of two 16-bit numbers in one BASIC integer because it only allows 31-bit positive numbers. This limitation does not exist in ARM assembler where the interpretation of bit 31 and of overflow are under programmer control.

I originally developed routines in assembler for the BBC-B where the word length was 8 bits. An early task, after acquisition of an Archimedes A310, was

to rewrite the routines in ARM assembler. The speed improvement was highly gratifying as was also the convenience of the longer word in relative addressing of more than 256 bytes. The techniques used and the results achieved may be of interest to some readers and while I hesitate to suggest a 'numbers' forum, some interchange of ideas might be welcomed.

Number representation

As hinted above, a key design decision is how the numbers are to be stored. My choice was binary and for the moment, only positive numbers are handled. They are stored just like BASIC integers – least significant byte first, but not limited to four bytes. Negative numbers could be represented in two's-complement or as positive integers with a sign bit. This decision has not yet been made – there are arguments for both. Variables are declared by DIM statements such as: DIM a% 255 which is an integer of 256 bytes maximum size. This would be a decimal number of some 600 digits. All variables must be declared before use.

Functions

There are currently five elementary functions, namely addition, subtraction, multiplication, division and comparison. The division function produces DIV and MOD simultaneously. The functions must be told the maximum length of the variable (not the actual) before being invoked, since the length is not implicit in the variable name.

Method of calculation

Addition and subtraction are carried out one 4-byte word at a time starting with the least significant word and propagating the 'carry' from each word to the next.

Multiplication is based on a core procedure for forming a 64-bit product of two 32-bit words. The routine calculates all such inner products and adds them together in appropriate relative positions to form the double-length product.

Division is by shift and subtract, just like school-book long division, but in binary. This is slow because a subtraction and test has to be performed

for as many bits as the dividend is longer than the divisor. (There has to be a faster method – I hope someone will tell me).

Use of CALL

The routines were initially designed to be invoked by commands such as: 'CALL product, multiplier, multiplicand', the length being communicated via one of the A%–H% system integers. This is easy to use but the long succession of CALLs, all of which look alike, makes programs much longer and obscures the mathematics. For this reason I tried an alternative.

Interpreter

The objective was to continue to use BASIC type expressions for long arithmetic e.g. $a\% = b\% * c\%$ rather than CALL mult,a%,b%,c%. The first foolish thought was that it might be possible to discard the real arithmetic in the BASIC interpreter and substitute long integer arithmetic. The obvious advantage was to have all the lexical and syntactical interpretation and fetching of the operand addresses for free. It was easy to dis-assemble the interpreter and examine it using *MEMORYI but understanding all of its ramifications was a different matter entirely. Without a fully described source text there would be endless pitfalls. I soon abandoned this idea in favour of a less ambitious scheme.

Permissible arithmetic statements were confined to those of the form: <answer>=<first operand><operator><second operand> where operands obey the same rules as BASIC integers and operators are confined to + - * / for arithmetic and < > for comparison. A BASIC literal integer is allowed as one operand. If no valid operator is found, a simple assignment such as $x\% = y\%$ is assumed. The need to quote four operands for division is avoided by having a common repository for the modulus.

Interpretation of IF statements and the like looked tricky so was not attempted. Instead, a statement such as $a\% = x\% < y\%$, where a% is a BASIC integer and x%, y% are long, gives a result a% such that $SGN(a\%) = SGN(x\% - y\%)$. This result can be tested by a BASIC command.

There has to be a means of indicating which parts of an application program are to be interpreted as

'long' and which as ordinary BASIC. It is done this way: you type RUN in the normal way and the program is executed as BASIC until it reaches 'CALL long,aux%,length%' whereupon the long interpreter takes over. It interprets the ensuing lines as 'long' until it finds a RETURN command, it then returns to BASIC. (The parameters aux% and length% are the division remainder (MOD) word and the arithmetic length respectively.)

At the present stage of development, error checking is minimal and disaster ensues from disobeying the rules.

Speed

The execution speed of the routines naturally depends on the arithmetic length. If length is N bytes, speeds are roughly $O(N)$ for add, subtract and compare and $O(N^2)$ for multiply and divide. The following approximate times were obtained with random numbers and lengths of 256 or 512 bytes:

512 byte add/subtract	55-65 μ secs
512 byte comparison of equals	47 μ secs
256x256 byte multiply	0.068 secs
512/256 byte divide	0.67 secs

The long interpreter takes about 20 microsecs for a typical instruction which is included in the above figures. This would lengthen with more error checking. It is a significant overhead for the add/subtract times.

The existing routines are not highly optimised and speed improvements are no doubt possible. Division is an obvious candidate for a better algorithm.

The future

Multiple length arithmetic using BASIC syntax presents an obvious problem in identifying variable length. Strongly typed languages such as Pascal and Ada declare the type, length and precision of their variables explicitly. You probably only need three lengths, 4-bytes standard BASIC, 'long' with a declared number of bytes and 'double length' for use in multiplication and division. Explicit declaration would allow more syntax checking at the cost of speed.

Routines could be faster if they knew on entry how long the actual variables were. In fact, the length of products and quotients are known exactly when

calculated and remainders must be shorter than divisors. Currently this useful information is discarded and actual length is re-determined whenever needed.

Extension to negative numbers has already been mentioned. A separate sign bit would probably be the easiest to implement, and probably the fastest. Decimal input/output is another obvious addition,

although very often one does not want to know the decimal representation, only that the number exists and its properties. What do you do with a 600 digit answer anyway?

I hope that this has provoked some thought and that anyone who has found it interesting will write to me with their views: 46 Hughes Road, Hayes, Middlesex, UB3 3AP. **A**

COPS – Adventure Game

Richard Forster

Is there, I wonder, anything wrong with the phrase "Adventure Game"? Is it now old hat, a statement of times past? I ask this because the first things that caught my eye when I received Alpine Software's Cops were the words "An Interactive Story". Taking a quick glance around I reckon that Infocom started the trend, so it seems to be an American influence, an attempt to make the games more acceptable. Call me old-fashioned, but I much prefer to play "Adventures" rather than "Interactive Stories".

This is not the only trend that has appeared in the world of Adventures. In the beginning, adventure games were original. Maybe this is not the correct word to use, as certainly themes like 'search the area for treasures', have been used many a time. What I mean is that we have seen the emergence of games based on something. You can now play 'the game of the film of the book' and so on.

The intro

There is a reason for this diversion. Upon loading up Cops (not an easy task) I was greeted with a picture of a police vehicle and the Hill Street Blues music. The music was pleasant, and I sat back enjoying it, thinking of the old series. Once the game begins, all similarity vanishes, with the exception of the puns on names – for example, you work around 'Hall Street'.

This is fine – after all, you want adventures to place you in another world as another person, but is it really fair to have a game giving a misleading appearance? Choose for yourself, after all there is always the saying "Don't judge a book by its cover". On these words let us bypass the cover and get right on inside to the 'story'.

The plot

Cops puts you in the roll of a rookie policeman. You and your partner, Rob-O-Cop, are assigned to a grade school to investigate a simple robbery. As the back of the game boasts, things rapidly go from bad to worse, in more ways than one. Along with this problem, the Police Chief has been kidnapped.

Everything seems fine at the start. You are briefed, told you are on the robbery case, and that everybody else is on the Police Chief case. initially, I took this to mean I was not on the missing Chief case, but how wrong I was. In order to get evidence for your case, you are forced to go and investigate the Chief's house and to play, what appeared to me, a game of chance.

I do not know about you, but I like logical progression in a game. Still, you cannot have everything, after all the game does tell of some pretty impressive features : High resolution graphics screens and a very advanced parser, to name but two. With items like this at our fingertips who needs logic?

Graphics

The graphics in the game were, to say the least, disappointing. I have been spoiled by Magnetic Scrolls; having played their adventures, other games graphics come a poor second but these are not as bad as some I could mention! The pictures do contain the occasional useful and subtle detail and the two digitised pictures, while they would have been better in mode 15, look quite acceptable from a distance.

The parser

The parser will be familiar to anybody who has used ALPS. There are several good points about the parser and, while not as good as Magnetic Scrolls',

it does cope with sentences and multiple commands. Unfortunately, it does have a habit of misunderstanding. For example, single command sentences are often thought to be two separate commands. The exclusion of multiple commands might have eradicated this problem.

User input

Input into the game is very convenient, however. As well as normal keyboard text input, the mouse can be used. At the bottom of the screen is a set of buttons with common commands, which can be accessed by the mouse. The second, more powerful, feature is that, by pointing at a word anywhere on the screen and clicking a button, the word is entered into your command. This saves a large amount of typing and it is worthwhile getting into the habit of using it.

Character interaction

There is limited character interaction in the game. You can ask the various characters about people and items. You can also ask them to do things. This is where the parser falls down occasionally because, if you ask somebody to do something, it occasionally makes you do part of the request after telling you the character's response to the first part.

Puzzles

Puzzles play a major part in any adventure, so it is worth looking at the puzzles found in the game. Most of them which I have encountered seem fair, though there does seem to be the occasional random part. Some lateral thinking is required, though the game is not immensely hard once you have realised you have to visit the Chief's house. There are one or two "sudden death" situations, but you quickly learn to avoid them and a RAMSAVE feature is provided.

The game comes in a sturdy plastic box, a departure from the normal blue and white cardboard one that Alpine Software usually use. The instructions are helpful, and give you enough information to get you started. If you send them a large s.a.e., a hint sheet is available and, though I have not seen it, if Plague Planet is anything to go by, it will be excellent.

ALPS

The game was written using ALPS, the adventure generating system, and is a good demonstration of

what can be accomplished. If you possess ALPS, you can load in the data files and explore a bit in the program, seeing how various bits were encoded. I had some initial problems running the game, it seems to need to be run from the desktop and I usually use BASIC. If you have ALPS, it gives a convenient way of running the game from within its own environment.

Conclusion

Overall, the game is playable, but nothing special. It is not as good as Plague Planet, and at times the parser seems to be struggling. If the graphics are viewed as an added extra, and if the game is persisted with, you are left with an acceptable adventure with some nice touches. It would not be my first choice as a game, but then again it would not nearly be my last. **A**

Archimedes

Expansion Card

Dual User Port

Connect your ARC to the outside world!

This module provides the Archimedes 300 or 400 with two 8 bit (or one 16 bit) bidirectional input/output ports with handshaking and two hardware timers. Uses the 2MHz 65C22 VIA.

- ✓ BBC Micro User Port Compatible
- ✓ Simple to use from Basic, C, ARM using SWI calls or *FX 150/1 (Sheila)
- ✓ Control routines supplied in ROM
- ✓ Interrupts supported (IRQ or FIQ)

Dual User Port
Expansion Card
only £84.95 + VAT

Official Government and
Educational Orders welcome.

Other products coming soon ...

**Science
Frontiers**

7 Port Hill Court
Gallowgate
Aberdeen
AB1 1DU



Tel: 0224 642765 pm

Writing RISC-OS Applications – 3

Alexander Goh (Sandie the Walrus)

I gather from various letters I have received and from talking to people at the CIS Archimedes Spring Fair that many of you still do not have the RISC-OS PRM and are trying to program the machine without it. I really don't think it is realistic to try to do that.

I could list all the SWIs and the exact format of the data blocks required by each one but it would take much more than a series in Archive to do it due to the sheer volume of information involved. Couple this with the lengthy application notes and introduction given in the PRM and you have 205 pages of text (pp 1125 – 1130)!

The manuals are essential for writing anything more complex than a few lines of BASIC and although you might think them a little bit expensive, you should realise that you get four volumes each as thick as the User Guide (~400 pages) and a 60 page index.

You end up paying about £17.50 per volume, which is quite reasonable considering that the BBC BASIC Guide is £19.95. Further more, bear in mind that the same set of manuals for an Atari ST would cost you about £200 and the Macintosh equivalents are a mere £199.95 per volume.

Window co-ordinates

When dealing with windows, you have to play with three sets of co-ordinates, all of which use the same units but with different origins. These are called "OS co-ordinates" and a screen contains 1280 by 1024 such units. However, in most modes, all references to co-ordinates must be multiples of 2 (x co-ords) and 4 (y co-ords) because only about 640 by 256 pixels can be displayed on a standard monitor.

Screen co-ordinates, as used for the MOVE and DRAW commands in BASIC, have their origin at the bottom left corner of the screen increasing to (1280,1024) at the top right (unless you're in a non-standard mode like 16).

This system of screen co-ordinates is also used for the window visible area co-ordinates, which define the size and position of the window on the screen. To place a window of size 400 by 200 centrally you

would therefore display its bottom left corner at (440,412).

Window co-ordinates are used to specify locations inside a window's work area and start with the origin at the top left of the work area. To further confuse matters, the y co-ordinates increase negatively (logical but weird) as you proceed down the window towards the bottom.

Finally, there are the scroll offsets which determine which part of the work area is represented by the visible area. Scroll offsets start at the top left and increase in the same direction (negatively downwards) as the work area co-ordinates. If this sounds rather confusing follow this little "tutorial":

Imagine that the screen is 1280 by 1024 OS units (which it is), and that a character is 16 by 32 OS units (also true). This means that we can, in theory, get 80 lines of 32 characters on the screen in a window. In practice, the title bar, etc would reduce this somewhat but I am ignoring them for the purposes of this example.

If we had an Edit file containing 64 lines of 80 characters and we displayed this in a window, the window's work area would be 1280 by 2048 OS units in size and therefore be twice the height of the screen. The window co-ordinates of the top left of the work area would always be (0,0) and the bottom right always (1280,-2048), but the actual screen co-ordinates of these points would vary depending on how we positioned the window's visible area. This makes re-drawing complicated as we have to get the window co-ordinates of each object to be redrawn, but print them at the screen co-ordinates.

If we placed the window at the top left of the screen, the screen co-ordinates of the top left character would be (0,1024) but its window co-ordinates would still remain unchanged at (0,0). The scroll offsets mark the amount by which the window has been scrolled and therefore make the situation worse as we now have to take the scrolling of the visible area into account. For example, if the above window was scrolled 32 OS units down, to start displaying at the second line of text, the screen co-ordinates of the top left of the work area would change to (0,1056).

A full understanding of this co-ordinate system is required and can be obtained from experimenting with Edit windows or by playing with suitably shaped pieces of graph paper on which the co-ordinates have been written. This is covered in the PRM, pages 1130 to 1134, where the following formulae also come from:

```
work area x = screen x - (visible
    area min x - x scroll offset)
work area y = screen y - (visible
    area max y - y scroll offset)
```

Summary

Work area is the total size of the window which may exceed that of the screen. The origin is at top left, so y co-ordinate decreases as you go down the window. Work area co-ordinates are used for positioning things within windows and never change.

Visible area gives the size and position of the window as it appears on the screen, using screen co-ordinates with the origin at the bottom left. As graphics operations use screen co-ordinates, to redraw a window it is necessary to translate the work area co-ordinates of the object in question to screen co-ordinates before plotting it. Visible area co-ordinates change as the window is moved and re-sized.

Scroll offsets: these effectively mark which co-ordinate of the work area can be seen at the top left of the visible area. If the window cannot be scrolled or will always be scrolled to show the top left of the visible area, then scroll offsets will be 0 and can be ignored – otherwise they must be figured into your re-draw.

Maintaining windows

Even when a window has been created, it will not actually appear on the screen until it has been opened. This is done via the Wimp_OpenWindow call which takes a block containing all the necessary information in R1.

```
0  window handle
4  visible area minimum x co-ordinate
8  visible area minimum y co-ordinate
12 visible area maximum x co-ordinate
16 visible area maximum y co-ordinate
20 x scroll offset
24 y scroll offset
28 handle to open window behind (-1 top, -2
    bottom)
```

By specifying the values in +4 to +24, you can open the window at any size and position on the screen. The value in +28 determines where the window will be opened in relation to the other windows. If it is the handle of another window, your window will be placed directly behind it, otherwise it will be placed at the top of the pile if -1 and the bottom if -2.

Sometimes, as when responding to an Open_Window_Request, this block is already set up for you, in which case you can call Wimp_OpenWindow straight away. At other times you will need to set the block up yourself, which is where Wimp_GetWindowState comes in handy. This SWI is passed a block containing a window handle at +0 in R1 and fills in the rest of the block by interrogating the Wimp. The block contains the same information as is required to open a window.

It is needed when opening a window which is not already on the screen, or it can be used to obtain the co-ordinates of a window's visible area to perform a graphics operation such as a redraw. Calling it excessively can cause problems as it will return the old position of a window in response to an Open_Window_Request from Wimp_Poll, so write the program without it and add in calls until it works and you gradually learn when to use it. Generally though, call it when opening a window that is not already on the screen or when you need to know the window's co-ordinates.

To move, re-size or change the depth of a window, re-open it in a new position/state. To close it, use SWI Wimp_CloseWindow which takes a block in R1 containing the handle of the window to close at R1 +0.

Redrawing windows

As I have already said, you can get the Wimp to do this for you, but if you want to do it yourself, switch off the auto-redraw bit of the window flag. Then anytime the window needs to be redrawn, Wimp_Poll will return a Redraw_Window_Request (reason code 1) which you should respond to as follows. (Note that you must do the required redraw operation before using any other Wimp routines and before you return to Wimp_Poll.) In the example below, *block* is the block passed to the Wimp_Poll SWI.

```

SYS "Wimp_RedrawWindow",,block TO
more
WHILE more
Redraw contents of window
SYS "Wimp_GetRectangle",,block TO
more
ENDWHILE

```

When a window needs to be redrawn, the Wimp splits the invalid area into a series of rectangles and sets a graphics window to the size and position of each rectangle. Wimp_GetRectangle gets each rectangle in turn, sets up the graphics window, puts its co-ordinates in the block and sets a flag to TRUE if another rectangle needs to be done afterwards – otherwise it returns FALSE.

When you re-draw the window, the invalid area is cleared to the background colour. You can either then re-plot the whole thing and rely on the graphics window to mask out any unwanted graphics, or interrogate the block to find out which parts of the rectangle need doing and redraw only those sections, which is obviously a lot quicker.

The format of the block is the same as that required for opening a window. If you forget to re-draw properly, your program goes into an endless loop. The Wimp says Redraw, you don't, so it asks again, and you don't, etc. Remember not to mask out redraw requests from Wimp_Poll if you have manually updated windows. More on redrawing later.

Creating icons

I haven't covered creating windows because this can be done using FormEd. Icons, however, often need to be created or changed and you cannot expect someone to have to resort to FormEd in the middle of using an application. An icon has a 32 byte block:

- 0 minimum x co-ordinate
- 4 minimum y co-ordinate
- 8 maximum x co-ordinate
- 12 maximum y co-ordinate
- 16 icon flags
- 20 12 bytes of icon data

The icon flags are detailed on pages 1180 to 1187 of the PRM – there are just too many for me to list here. The icon data can be either 12 bytes of text or a sprite name or they can be pointers to a longer string. In this case the icon is said to be indirected and bit 8 of

the icon flags is set. If this is the case, then the last 12 bytes of the block are as follows:

- 20 pointer to text string or sprite name
- 24 pointer to validation string (-1 if none)
- 28 length of buffer for items in +20

As you can see, this system allows you much more freedom in using icons. I hope to return to the subject of indirection and validation later.

To create any icon, Wimp_CreateIcon is used but the block has an extra entry inserted at +0 which moves all other entries down by 4 bytes. This is the handle of the icon's parent window: -1 for the right hand side of the icon bar (applications), or -2 for the left (devices and system utilities). Also, all co-ordinates on the icon bar should be relative to the icon and not any set of window co-ordinates.

The little routine below creates an icon for the icon bar from a 34 x 17 pixel sprite called "ledit". In other words, it creates an icon identical to the one produced by the Edit application when it is running. Note that I have assumed that the icon will not be created in a non-square mode like 16 so have used scale factors of 2 and 4. More on mode independence later.

```

REM Create on right of icon bar
block!0 = -1
REM x and y min are 0 for icon bar
block!4=0 : block!8=0
REM x and y max are 34x2 and 17x4
block!12=68 : block!16=68
REM icon flag (sprite, allow
mouse clicks) & sprite name
block!20=&3002 : $(block+24)=
"!edit"
REM Create and store handle in
suitable variable
SYS"Wimp_CreateIcon",,block TO
iconbar_handle

```

Finally, I've received a letter from Leslie Cleave saying that window identifiers should be longer than 3 characters and please could I warn people that things go unpredictably wrong otherwise. I haven't had time to check this yet as I received the letter too late, but I'll have a look at the problem by next month. **A**

Teletext on the Archimedes

Richard House

Little has been published about Teletext on the Archimedes but from Archive 'Help' requests, there are some people who are interested in the subject. My aim in this article is to describe how teletext information can be obtained in an Archimedes useable form.

Firstly, it is worth clarifying what teletext is for those unfamiliar with the subject. Teletext data is transmitted with the television pictures you receive on your TV. Those with suitable TVs can display this data on their TV screen. The advantage of receiving teletext data by computer is that the data can be manipulated and stored. It is not possible to transfer teletext information from a TV set to a computer.

It is worth emphasising that using teletext with a computer involves no connection to telephone lines and so the only cost, after purchase of an adaptor to receive the transmissions, is that of a normal TV licence. Each TV channel broadcasts teletext data. Until August 1989, BBC2 transmitted software and weather satellite picture data which could be downloaded but this unfortunately ceased due to an enhancement of the normal CEEFAX service. However, there remains much useful information on all 4 channels such as news headlines, stock market information and TV programme schedules to name but a few.

The old BBC machines were specifically designed to support teletext mode graphics and there have been a number of adaptors available for capture of teletext data on the old BBC micros. The teletext mode on the Archimedes has been provided probably more to retain compatibility than for any other reason.

The provision of adaptors for the Archimedes has been limited and the only ones currently being advertised are by Morley and Ground Control. The Morley adaptor connects directly to the podule socket and the Ground Control adaptor looks like the version produced by SCML which connected to the printer socket. I have had the old Acorn Teletext adaptor working on my BBC B for the last 3 years

and have recently written a 5 kbyte module to drive it on the Archimedes from the I/O podule 1MHz bus with ATS (Advanced Teletext System) compatible commands.

The requirements are a good TV signal, an adaptor and software to drive the adaptor. Adaptors receive the TV signal and present all the teletext data to the computer in a raw form. The software controls the adaptor and provides high level data manipulation without the user worrying about the exact data format. Normally the adaptor and software are provided together.

Organisation of teletext data

Teletext transmissions are divided into pages which are essentially screens of information. These pages are numbered with 3 decimal digits – for example, on BBC2 CEEFAX, page 201 displays the financial headlines. The first digit of the number is called the magazine, so the example is page 01 from magazine 2. Each page is made up from a number of rows or broadcast packets. These packets are transmitted one after the other so there is an interval before a selected page appears from the adaptor. Pages can also have sub-pages which are transmitted in a cycle under the page number.

Data format

In each packet there are 42 bytes of data organised as follows.

1st byte: Magazine number (0 – 7, but 0 is interpreted as 8)

2nd byte: Packet number (0-31) (screen row)
bytes 3 – 42: 40 bytes of data

Packet 0 has information about the page number, sub-page number and some status flag bits and provides the header row at the top of the teletext screen. Packets 1-24 provide the remaining screen rows. Packets greater than 24 contain other non-display information such as a cyclic redundancy check and the link or Fastext page numbers.

The data bytes represent the mode 7 teletext character set, duplicated in the ranges 0 – &7F and &80 – &FF. The character "A" can be either &41 or &C1.

Extracting the information

The processes needed to get information from teletext:

- a) Select the required page
- b) Load the page into memory
- c) Scan the page for the information required

With ATS software, star commands are available for stages (a) and (b) but more complex routines can be developed using the ATS OSWORD &7A calls. The Morley adaptor uses BASIC procedures to achieve the desired results.

A basic program structure using ATS commands would be:

```
DIM memblock 1200 ;allocate a 1200
                        byte memory block
*TTXON ;enable teletext adaptor
*BBC2 ;select BBC2 channel
*PAGE 221S0001 ;set adaptor
                        searching for magazine 2,
;page 21 sub-page 1
OSCLI("TRANSFER "+STR$(memblock))
                        ;transfer selected page
                        from adaptor
;to memory block
*TTXOFF ;disable teletext adaptor
OSCLI("**SAVE P221S1 "+STR$(
                        memblock)+" +1200")
;save page to disc
```

Note that using OSCLI avoids having to be concerned with the actual value of memblock.

Writing the software to extract the information from the page now stored in memory requires a knowledge of the page layout. This can be easily be determined by displaying the page on the screen and counting down to the required row and then in, to the required character column position. Luckily, the page layouts do not change often. The packets (screen rows) are stored in 42 byte blocks with the 2nd byte indicating the packet number (screen row). Blank rows are not always transmitted and therefore may not be stored in the memory block. To find a row involves looking at the 2nd byte in each 42 byte block. If it is found, the column position can be reached by counting in the required number of bytes.

Extracting the information from any part of a row involves reading the bytes into a string. As this is

done it is worth ensuring each byte is in the normal ASCII character range &20-&7F to avoid any problems outside the teletext mode 7. Some conversion of numbers may be necessary as teletext allows representation of fractions "1/4", "1/2" and "3/4" by single character bytes. The number of character bytes read into the string can be fixed from the knowledge of the page layout or set to recognise any control character byte after the desired string. Liberal use of these control characters is made by the teletext authors to obtain the different colours and graphics on the screen. The control character appears as a blank space but affects the interpretation of the bytes following on that row.

The BASIC program with this article extracts the share prices of companies listed on page 221S0001 (page 221, sub-page 1) on CEEFAX and stores them in arrays. The shares are listed on rows 5-19 in the following format:

Column		
Position	Byte	Meaning
0	&86/83	blue or yellow alpha control character
1-7		share name
8	&07	white alpha control character
10-13		price in pence
14		possible fraction
15	&86/01	blue alpha or red alpha
16	&AB/AD	"+" or "-" char
17-18		price movement in pence
19		possible fractional element
20-39		repeat of 0-19

The teletext page and the extraction program should be found on the monthly disc. The program first displays the page in Mode 7 and then extracts the share names and prices into arrays for printing out on a Mode 12 screen.

```
10 REM >TTXextr
20 REM Program Teletext data
                        extraction demonstration
30 REM Version 1.0
40 REM Author R M House
50 REM Date 30 Jan 90
60 :
90 REM initialise all variables
100 pagesize=1200
110 DIM name$(30),price(30)
120 DIM pagemem pagesize
```

```

130 :
140 REM load T'text page into memory
150 OSCLI ("*LOAD P221/1 "+STR$
~pagemem)
160
170 PROCdisplaypage (pagemem)
180
190 MODE 12
200 REM extract data from page
210 row=5
220 J=0
230 WHILE row <=19
240 rowstart=FNfind(row,pagemem)
250 IF rowstart<>0 THEN
260
270 REM first half of row
280 name$(J)=FNreadname(rowstart)
290 price(J)=FNreadprice(rowstart)
300 J=J+1
310 REM second half of row
320 name$(J)=FNreadname(rowstart+20)
330 price(J)=FNreadprice(rowstart+20)
340 J=J+1
350 ENDIF
360
370 row=row+1
380 ENDWHILE
390 REM Print out arrays on screen
400 J=J-1
410 PRINT "SHARE","PRICE (PENCE)"
420 FOR I=0 TO J
430 PRINTname$(I),price(I)
440 NEXT I
450 END
460 :
470 REM *****PROCEDURES*****
480 DEFFNfind(row,address)
490 LOCAL I
500 I=0
510 pkt=address?(1+(I*42))
520 WHILE I<30 AND pkt<>row
530 I=I+1
540 pkt=address?(1+(I*42))
550 ENDWHILE
560 IF pkt=row THEN =(address+2+
(I*42)) ELSE =0
570 :
580 DEFFNreadname(start)
590 LOCAL name$,I
600 name$=""
610 FOR I=1 TO 7
620 name$=name$+CHR$(start?I AND&7F)
630 NEXT I
640 =name$
650 :
660 DEFFNreadprice(start)
670 LOCAL price$,frac,fraction,I
680 price$=""
690 FOR I=10 TO 13
700 price$=price$+CHR$(start?I AND
&7F)
710 NEXT I
720 frac=start?14
730 CASE frac OF
740 WHEN 220: fraction=0.5
750 WHEN 251: fraction=0.25
760 WHEN 253: fraction=0.75
770 OTHERWISE: fraction=0
780 ENDCASE
790 =VAL(price$)+fraction
800 :
810 DEFPROCdisplaypage (pagemem)
820 REM routine to display Teletext
page
830 MODE 7
840 colstart=8 :REM start at column
8 for first row
850 PRINTTAB(colstart);
860 row=0
870 REM for each screen row
880 WHILE row<24
890 REM find start of row in memory
900 rowstart=FNfind(row,pagemem)
910 IF rowstart<>0 THEN
920 FOR col=colstart TO 39
930 char=(rowstart?col)OR &80
940 VDU char
950 NEXT col
960 ELSE
970 REM if not found, print blank
line
980 VDU 10,13
990 ENDIF
1000 colstart=0
1010 row=row+1
1020 ENDWHILE
1030
1040 PRINT"Press any key to
continue...";
1050 A=GET
1060 MODE 12
1070 ENDPROC

```

Summary

I hope that the technical aspects of this article have not put anyone off using teletext. It is very simple – just look at it displayed on the screen even if you do not want to extract information. I have found that it provides a dynamic challenge in programming as well as providing useful information. I currently extract 309 pieces of share information daily for

charting purposes and the only saving grace the computer has with my wife is the production of a printed TV programme listing for the evening! Other uses could include the printing of all those TV programme recipes broadcast on CEEFAX pages or the production of the latest news broadcasts in printed form. **A**

Z88 File Transfer

David Holden

These programs have been written to enable the Archimedes to be used as an accessory for the Z88 portable. (The tail wagging the dog?!)

The Z88 can communicate with the world only through its serial port and cannot save files except to its internal RAM or EPROM. It is possible to link it to a serial printer but if you have an Archimedes, your printer probably has a parallel interface. This will require an adaptor for the Z88. Even if you buy the adaptor, you still have the aggravation of switching leads and the further problem that some Z88's will insist upon sending a linefeed at the end of every line – the opposite of the default setting on the Archimedes – so you must also alter your printer's configuration.

The programs

There are four programs in this series which are designed as transient programs to be run from within the desktop. I will describe the first three here and the fourth in a later article.

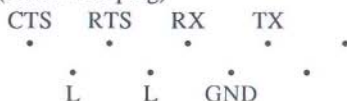
The first redirects the characters received by the Archimedes serial port from the Z88 to the printer, allowing the Archimedes to act as a dumb link between Z88 and printer. The next two enable the Archimedes to act in a similar way as a buffer between the Z88 and the disc drive, letting you save and load files to the Archimedes discs. The fourth program is a combination of all three with the addition of menus to make disc operations easier and a facility to send and receive multiple files to let you save all your Z88's memory.

The hardware

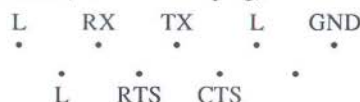
Before you can use any of these you will need a lead to connect the Z88 to the Archimedes. You can easily make this from two 9 pin 'D' plugs, one male

and one female, and a short length of cable. The cable needs five wires and could be screened four core cable as sold in Hi-Fi shops, using the screen as Ground, or the cheaper 6 core flex sold for burglar alarms. The two plugs should be wired as follows:

Z88 (the MALE plug)



Archimedes (the FEMALE plug)



Inside each plug you must link together three pins marked 'L' on the Archimedes and two marked 'L' on the Z88.

These diagrams show the plugs looking on the solder lugs, i.e. as you see them when soldering the leads. To make a lead connect:

GND	to	GND
RTS	to	CTS
CTS	to	RTS
RX	to	TX
TX	to	RX

The programs may be run from the OS or BASIC prompt or from the desktop, which is the simplest method.

Print program

To use the Z88->Print program, double-click on the program icon and the usual window will open in the centre of the screen telling you that the program is ready. Now set the Z88 to print. The easiest way to do this is to load the file into Pipedream and press the square key followed by <P> and <O>. The Z88

thinks it is sending the file direct to the printer and so includes all the formatting codes which the Archimedes passes on to produce the document. When you have finished printing your file(s), press <escape> to quit the program.

Filing programs

The disc operations use the Z88's Import/Export mode. To save a file to disc, double-click on the Z88->Disc program. On the Z88, press the square key followed by <X>. Press <S> for Send followed by <enter>. Now enter the name of the file you want to send and press <enter> again. You should see the number of bytes received being counted by the Archimedes. When the end of the file has been received, you will be prompted for a filename and the file will be saved to disc. This method was used rather than extracting the name from the Z88 file as you may wish to keep more than one version of the file and extracting the Z88 name would overwrite the earlier version.

To send a file to the Z88, double-click on the Disc->Z88 program. Again on the Z88 press the square key and <X>, press <R> for Receive and enter the filename. At the Archimedes' end, enter the name of the file you wish to send. You will see the bytes counted as they are sent and the Z88 will display the number of blocks received.

The default setting for the serial port on the Z88 uses the same settings as the Archimedes and is set to 9600 baud. All these programs use 9600 baud as this has been found reliable with RISC-OS or Arthur (with the serial port fix). If you have changed any of these settings then you can either restore them or add extra lines to the programs to configure the Archimedes serial port to your preferences. The baud rate is set by the variable *speed* which can be seen at the beginning of each program. If you wish to change this, see page 311 of the User Guide for the various baud rates and their numbers.

The disc programs contain two other variables which you may wish to change. Both of these programs use a buffer to hold the files as this is much faster and a lot kinder to the drives than BPUT and BGET. I have set the size of this to 24k but you can alter it to any size you like by changing the variable *bufferize* to the number of k you require.

The other variable is the directory prefix, *path\$*. This is set to "@" at present which means load and save files to the current directory. This can be changed to put them in the drive and directory of your choice.

```

10 REM >Disc->Z88
20 REM Transfer files from Disc - Z88
30 REM (C) D.J.Holden Feb.1990
40 buffersize=16 :REM Maximum size
                        of file in Kilobytes
50 speed=7 :REM Serial transfer
                        speed, 9600 baud
60 path$="@" :REM Drive/dir files
                        will be loaded from
70
80 DIM buffer% (buffersize*400),
                        filename% 64
90 SYS"OS_Byte",8,speed
100 SYS"OS_Byte",7,speed
110 *FX21,2
120 REPEAT
130   PROCload:PROCSend
140 UNTIL FALSE
150 END
160
170 DEFPROCload
180 LOCAL ERROR
190 ON ERROR LOCAL IF ERR=17 END ELSE
        PROCreport:PROCload:ENDPROC
200 PRINT'"CHR$11" Enter file name
        ->";
210 SYS"OS_ReadLine",filename%,64,33
        ,126 TO ,length%
220 IF length%=0 END
230 SYS"OS_File",&FF,path$+"."+
        $filename%,buffer%,0 TO
        ,,,,length%
240 ENDPROC
250
260 DEFPROCsend
270 LOCAL ERROR
280 ON ERROR LOCAL PROCreport:ENDPROC
290 PRINT'CHR$7" Press SPACE when
        Z88 is ready to receive"'
300 REPEAT:K%=GET:UNTIL K%=32
310 PRINTCHR$11" Bytes sent -";
320 ptr%=0
330 REPEAT
340   SYS"OS_SerialOp",3,buffer%?ptr%
350   ptr%+=1:PRINTTAB(15,VPOS);ptr%;
360 UNTIL ptr%>length%
370 ENDPROC
380
```

```

390 DEFPROCreport
400 *FX2,0
410 PRINT:PRINTCHR$7"Error! ";:REPORT
      :PRINTERL
420 PRINT:PRINT"      Press SPACE to
      continue."
430 *FX21,0
440 REPEAT:K%=GET:UNTIL K%=32
450 ENDPROC

10 REM >Z88->Disc
20 REM Transfers files Z88 to disc
30 REM (C) D.J.Holden Feb.1990
40 buffersize=24 :REM Maximum size
      of file in Kilobytes
50 speed=7 :REM Serial transfer
      speed, 9600 baud
60 path$="@ " :REM Drive/dir. files
      will be saved to

70
80 DIM buffer% (buffersize*&400),
      filename% 64
90 SYS"OS_Byte",8,speed
100 SYS"OS_Byte",7,speed
110 REPEAT
120 PROCreceive
130 *FX2,0
140 IF ptr%>2 buffer%?ptr%=69:ptr%
      +=1:PROCsave ELSE END
150 UNTIL FALSE
160
170 DEFPROCreceive
180 LOCAL ERROR
190 ON ERROR LOCAL IF ERR=17 ENDPROC
      ELSE PROCreport:ENDPROC
200 *FX15,0
210 PRINT'TAB(16)"_____
      "
220 PRINT'TAB(16)"READY TO RECEIVE
      FROM Z88"
230 PRINTTAB(16)"_____
      "
240 PRINTTAB(20)CHR$11"Bytes received
      -";
250 ptr%=0:*FX2,1
260 REPEAT
270 buffer%?ptr%=GET:ptr%+=1
280 PRINTTAB(37,VPOS);ptr%;CHR$13;
290 UNTIL I%=69 AND buffer%?(ptr%-2)
      =27
300 ENDPROC
310
320 DEFPROCsave
330 LOCAL ERROR
340 ON ERROR LOCAL IF ERR=17 ENDPROC
      ELSE PROCreport:PROCsave:ENDPROC
350 PRINT'" Enter file name ->";
360 SYS"OS_ReadLine",filename%,64,33
      ,126 TO ,length%
370 SYS"OS_File",10,path$+"."+
      $filename%,&288,,buffer%
      ,buffer%+ptr%
380 ENDPROC
390
400 DEFPROCreport
410 *FX2,0
420 PRINT:PRINTCHR$7"Error! ";:REPORT
430 PRINT:PRINT"      Press SPACE to
      continue."
440 *FX21,0
450 REPEAT:K%=GET:UNTIL K%=32
460 ENDPROC

10 REM >Z88->Print
20 REM Sends Z88 input from serial
      port direct to printer
30 REM (C) D.J.Holden Feb.1990
40
50 speed=7 :REM Serial transfer
      speed, default 9600 baud
60 SYS"OS_Byte",8,speed
70 SYS"OS_Byte",7,speed
80
90 ON ERROR PROCend:REPORT:END
100
110 PRINT'TAB(16)"_____
      "
120 PRINT'TAB(16)"READY TO RECEIVE
      FROM Z88"
130 PRINT'TAB(15)"Make sure printer
      is on line"
140 PRINTTAB(14)"_____
      "
150 *FX21,1
160 *FX2,1
170 *FX3,10
180 REPEAT
190 char%=GET
200 REPEAT:SYS"OS_Byte",128,252 TO
      ,free%:UNTIL free%>10
210 VDU char%
220 UNTIL FALSE
230
240 DEFPROCend
250 *FX3,0
260 *FX2,0
270 *FX21,0
280 ENDPROC

```

Using the PC Emulator – Part 1

Richard Forster

When I bought my Archimedes, I had never used an IBM PC. This was not something which really bothered me. "After all," I thought "what can such an out-moded range of machines offer me?" The only idea which came to mind was 'compatibility' (an infamous sales pitch) and it quickly left.

Today, having used an IBM compatible (albeit in the form of Acorn's excellent PC emulator) I think differently. Of course, if it came to a straight choice between an Archimedes and a PC, the Archimedes would win easily but I would feel rather at a loss if I had to do without the PC programs I now use.

My knowledge of MS-DOS has increased considerably but, along the way, I found two main stumbling blocks to progress. The first was lack of software, though that was remedied by some PD (public domain) programs. The more serious was lack of information. When you consider the manuals which generally accompany software, Acorn's attempt to document the PC emulator is pretty feeble.

During the next few articles I hope to impart to you some of the useful knowledge which has come into my possession. With this, you can start down the road to manipulating the beast – a journey which will take us through configuration and compatibility to the more mysterious side.

In the beginning

Unlike the Archimedes, the PC has its operating system on disc and not in ROM. The upshot of this is that every time the PC resets, it is forced to re-load its operating system—MSDOS. For this to run, there are three vital files. Two of these are hidden and cannot be directly accessed – the third is called COMMAND.COM and is visible to all.

Whenever possible, you should backup your software. Making a duplicate of the original boot disc is highly recommended but it is not immediately obvious how to do this. You might have tried a *BACKUP command from the desktop, only to discover a disc error. It is important to remember that once the PC emulator is running you are in a different environment; everything is different including disc formats.

Follow Acorn's instructions to get the emulator up and running. Once you are at the prompt (A> or C>) type in:

```
DIR
```

You will then see the contents of your boot disc. You might think that you have been supplied with some software. In fact, the files you see are more like commands. Formatting is built into the Archimedes but the PC must run the program FORMAT.EXE to format its discs. The DIR command is one of the few 'internal' commands built into COMMAND.COM. CLS is another built-in command – try it!

In order to make a backup we will need to format a disc to the PC format. Looking through the manual we see an example of how to use it. Ignore this for now! Instead, write protect your original disc, get a blank one ready but don't put it in yet and type in:

```
FORMAT A: /S
```

When you have followed all the instructions the computer gives you and all the action is over, (a PC takes its time formatting a disc) do a directory listing. Notice our newly formatted disc has the file COMMAND.COM present. This is due to the /S parameter we added, making this a boot disc. The two invisible files are also present on the disc. When you next load the PC emulator, you can use this disc as the boot disc you must insert.

Now we need to copy some files onto the disc. The PC names the various drives in a different way from the Archimedes. Whereas the Archimedes uses numbers, the PC uses letters. This means that what was previously 0 is now A. A hard disc drive becomes C (and not D) and a second floppy is B. Drive B has another use however.

On a single floppy machine, the situation often arises when you want to copy between discs. It is not possible to get the machine to copy from A to A, but it is possible to copy from A to B. Using this latter form will allow you to complete the copying as the machine, realising there is only one drive, will prompt you to change discs.

Reference to these drives is also done in a slightly different fashion. To refer to drive A you use A:

(unlike the Archimedes' :0.) To change drive you simply type in the drive letter followed by the colon and the prompt will change e.g. C: changes to hard disc whereas typing:

```
C:TEXT.DOC
```

would view TEXT.DOC on the hard disc but without changing drive.

Now we can transfer some useful files onto our new boot disc. We could copy all of the files but some of them are pretty useless. Still if you want to make a full backup you should type:

```
DISKCOPY A: B:
```

For most uses, you need only copy a few selected files. The copy command is built into the file COMMAND.COM and, as such, does not require the MS-DOS boot disc to run. First we will copy the format command to our new disc. Type in:

```
COPY A:FORMAT.EXE B:FORMAT.EXE
```

Disc A should be the original and B the new one we have just formatted. The A: was not necessary in the above command as we are already on drive A. To demonstrate this, copy the next file as follows:

```
COPY CONFIG.SYS B:CONFIG.SYS
```

You might have noticed by now that PC files consist of a name followed by a full stop and a three letter extension. This is necessary and helps to add some extra description to a file. As well as this, they have a similar function to the filetype of an Archimedes file. So, if the extension is vital, how is it that we are running FORMAT.EXE by just typing FORMAT?

Some extensions have built-in uses on the PC. The three most important ones are .COM, .EXE and .BAT. A fourth extension, .SYS, is also going to occur a bit but it does not work in the same way. When you typed in FORMAT, the machine first searched for FORMAT.COM. Failing this, it looked for FORMAT.BAT and upon being once again unsuccessful it looked for FORMAT.EXE. This file was then executed.

This preference for execution does not continue into file manipulation. This is why we were forced to call it FORMAT.EXE when copying it. If we had a file called FORMAT.BAT and wanted to use FORMAT.EXE we would have to be more specific when specifying the file. It is a pleasant system (as

the extension is part of the name) and would allow say a file PROGGY.EXE and PROGGY.DOC to exist in the same area on the same disc.

Restart the emulator but this time using your new boot disc with its few files. Isn't the question about time and date annoying? This is a legacy from machines without RTC's (real time clocks) and the Archimedes, already possessing one, does not need the time and date entered all the time.

In order to remedy the situation we will create a new file. This file, called AUTOEXEC.BAT, is a special file which is automatically executed when the machine is booted.

We will ignore the text editor EDLIN.EXE until later. Instead we will use COPY, though in a different way from before.

When you boot up, if anything is found in AUTOEXEC.BAT, the time and date question does not appear. Therefore, if we do not want this question we just need to create the file. Even if it is empty it will serve our purpose. Our AUTOEXEC.BAT will consist of a blank line, followed by a ctrl-Z. This is because ctrl-Z is the end-of-file marker on a PC.

Type in:

```
COPY CON:AUTOEXEC.BAT
```

As far as the machine is concerned you are copying between two files CON: (i.e. the console or keyboard) and AUTOEXEC.BAT. In real terms, you are copying from the keyboard to a file. Simply press <return> then press <f6> (this key is set up to act as <ctrl-Z>) and press <return>.

To show you how this has worked, we will reboot the PC. Instead of reverting to RISC-OS and running the PC emulator again, press the following three keys together: <ctrl>+<alt>+<delete>. This is the PC version of a reset – be careful with it.

Well that about finishes it for now. Next month we will continue with the installation and examine a couple of the more useful files on the boot disc. **A**

Contact Box

• **Belgian Archimedes Users Club** contact Philippe Stroobandt (vice co-ordinator), Meidoornlann 17, 1851 Grimbergen, Belgium. **A**

Fact-File

(The numbers in *italic*
are fax numbers.)

- 4th Dimension
4mation
Abacus Training
Acorn Computers Ltd
Ace Computing
Active Sales
Akhter Computers Ltd
Aleph One Ltd (p11)
Alpine Software
Apricote Studios (p10)
Atomwide Ltd
Beebug
Cambridge International Software
Clares Micro Supplies
Colton Software (p18)
Computer Concepts (p30/31)
Computerware
Dabhand Computing
Dabs Press (p12)
EMR Ltd
Future Software (p3)
G.A.Herdman Educational
Ian Copestake Software (p29)
Intelligent Interfaces
Jim Markland (p15)
Lingenuity (Lindis)
Logotron
Micro Studio Ltd
Minerva Systems
Mitre Software
Morley Electronics
Musbury Consultants
Oak Computers (p4)
Relational Engineering
Science Frontiers (p49)
Silicon Vision Ltd
Simtron Ltd
Software Solutions
Techsoft UK Ltd (p16)
The Serial Port (p22)
Topologica
Vector Services Ltd
Watford Electronics
Wild Vision
P.O. Box 4444, Sheffield. (0742-700661)
Linden Lea, Rock Park, Barnstaple, Devon, EX32 9AQ. (0271-45566)
29 Okus Grove, Upper Stratton, Swindon, Wilts, SN2 6QA.
Fulbourn Road, Cherry Hinton, Cambridge, CB1 4JN. (0223-245200) (-210685)
27 Victoria Road, Cambridge, CB4 3BW. (0223-322559)
Studio Centre, 1 Ranelagh Gardens, London SW6 3PA. (01-384-2701)
Akhter House, Perry Road, Harlow, CM18 7PN. (0279-443521) (-442623)
The Old Courthouse, Bottisham, Cambridge, CB5 9BA. (0223-811679) (-812713)
P.O.Box 25, Portadown, Craigavon, BT63 5UT. (0762-342510)
2 Purls Bridge Farm, Manea, Cambridgeshire, PE15 0ND. (035-478-432)
23 The Greenway, Orpington, Kent, BR5 2AY. (0689-38852)
117 Hatfield Road, St Albans, Herts, AL1 4JS. (0727-40303) (-60263)
8 Herbrand Street, London, WC1N 1HZ. (01-278-6622) (01-837-6077)
98 Middlewich Road, Rudheath, Northwich, Cheshire, CW9 7DA. (0606-48511)
(-48512)
149-151 St Neots Road, Hardwick, Cambridge, CB3 7QJ. (0954-211472) (-211607)
Gaddesden Place, Hemel Hempstead, Herts, HP2 6EX. (0442-63933) (-231632)
11 Livestock Market, Hall Road, Norwich. (0603-507799)
5 Victoria Lane, Whitefield, Manchester, M25 6AL. (061-766-8423) (-8425)
22 Warwick Street, Prestwich, Manchester, M25 7HN. (061-773-8632) (-8290)
14 Mount Close, Wickford, Essex, SS11 8HG. (0702-335747)
10 Stokesay, Bidston, Birkenhead, L43 7PU.
43 Saint Johns Drive, Clarborough, Retford, Notts, DN22 9NN
10 Frost Drive, Wirral, L61 4XL. (051-648-6287)
43B Wood Street, Stratford-on-Avon, CV37 6JQ. (0789-415875) (-299600)
4 Shalford Close, Cirencester, Gloucester, GL7 1WG.
P.O.Box 10, Halesworth, Suffolk, IP19 0DX. (0986-85-476) (-460)
Dales Brewery, Gwydir Street, Cambridge, CB1 2LJ. (0223-323656)
22 Churchgate Street, Soham, Ely, Cambridgeshire.
69 Sidwell Street, Exeter, EX4 6PH. (0392-437756) (-421762)
26 Creechurch Lane, London, EC3A 5BA. (01-283-4646)
Morley House, Norham Road, North Shields, Tyne & Wear, NE29 7TY.
(091-257-6355) (-6373)
8 Fairhill, Helmsore, Rossendale, Lancs, BB4 4JX. (0706-216701)
Cross Park House, Low Green, Rawdon, Leeds, LS19 6HA. (0532-502615)
(-506868)
53 Severn Street, Middlesborough, TS4 2AY. (0642-613641)
7 Porthill Court, Aberdeen, AB1 1DU.
Signal House, Lyon Road, Harrow, Middlesex, HA1 2AG. (01-861-2173)
(-427-5169)
4 Clarence Drive, East Grinstead, W. Sussex, RH19 4RZ. (0342-328188)
Broadway House, 149-151 St Neots Road, Hardwick, Cambridge, CB3 7QJ.
(0954-211760)
Old School Lane, Erryrs, Mold, Clwyd, CH7 4DA. (082-43318)
Burcott Manor, Wells, Somerset, BA5 1NH. (0749-72234) (79794 data 300-2400)
P.O. Box 39, Stilton, Peterborough, PE7 3RL. (0733-244682)
13 Dennington Road, Wellingborough, Northants, NN8 2RL.
250 Lower High Street, Watford, WD1 2AN. (0923-37774) (01-950-8989)
6 Jesmond Road, Newcastle upon Tyne, NE2 4PQ. (091-281-8481) (-5150)

Norwich Computer Services 18 Mile End Road, Norwich, NR4 7QY. (0603-507057) (Fax 250095)

Archive

Subscription Magazine and Support Group for *Archimedes* Users

Archive Magazine contains:

- News
- Reviews
- Hints and Tips – a major feature
- Articles for Beginners
- The Latest Technical Information
- Program Listings
- Free Small Ad's Section
- HELP – Requested and Offered
- Contact Box – to help you form common interest groups

Bulletin Board: Ring the Archive BBS on 0603-745932, up to 2400/2400, scrolling, 8n1.

- Very Latest News
- Down-load Software
- Mailbox Facilities
- Chat line

Technical Help Service (£10 / year)

A telephone hot-line service for immediate help with your technical problems. Any member can send written enquiries, but for a fast response use the THS!

Members' Discount: Archive offers its members up to 15% off software and hardware from a range of different suppliers when purchased through Norwich Computer Services.

Subscription: 12 issues £17 (UK)
Europe £24, Middle East £29,
America / Africa £31, Elsewhere £34.
Technical Help Service £10

For more details: Ring Adrian or Alison on 0603-507057.

N.B. All earlier issues have now been re-printed – you may back-date your subscription as far as issue 1 (October 1987) – to take advantage of this huge bank of information.

Archimedes is a trademark of Acorn Computers Ltd.

* Please send copies of *Archive* magazine for ____ years starting from...
Volume ____ Issue ____

* Please enrol me on the Technical Help Service for one year. (£10)

I enclose a cheque for £ _____ payable to "Norwich Computer Services".

Name: _____

Address: _____

Postcode: _____

Norwich Computer Services, 18 Mile End Road, Norwich, NR4 7QY

Special three-year subscription starting from volume 1, issue 1 – £40 (UK)